

[illegible]

```
CCCCCCCC  RRRRRRRR  DDDDDDDD  RRRRRRRR  IIIIII  VV  VV  EEEEEEEEE  RRRRRRRR
CCCCCCCC  RRRRRRRR  DDDDDDDD  RRRRRRRR  IIIIII  VV  VV  EEEEEEEEE  RRRRRRRR
CC         RR      RR  DD      DD  RR      RR      II  II  EE      RR      RR
CC         RR      RR  DD      DD  RR      RR      II  II  EE      RR      RR
CC         RR      RR  DD      DD  RR      RR      II  II  EE      RR      RR
CC         RRRRRRRR  DD      DD  RRRRRRRR  II  II  EE      RRRRRRRR
CC         RRRRRRRR  DD      DD  RRRRRRRR  II  II  EEEEEEEE  RRRRRRRR
CC         RR  RR    DD      DD  RR  RR    RR  RR  EE      RR  RR
CC         RR  RR    DD      DD  RR  RR    RR  RR  EE      RR  RR
CC         RR      RR  DD      DD  RR      RR    RR  RR  EE      RR  RR
CC         RR      RR  DD      DD  RR      RR    RR  RR  EE      RR  RR
CCCCCCCC  RR      RR  DDDDDDDD  RR      RR    RR  RR  EEEEEEEEE  RR      RR
CCCCCCCC  RR      RR  DDDDDDDD  RR      RR    RR  RR  EEEEEEEEE  RR      RR
```

```
LL  IIIIII  SSSSSSSS
LL  IIIIII  SSSSSSSS
LL  II      SS
LL  II      SS
LL  II      SS
LL  II      SS
LL  II      SSSSSS
LL  II      SSSSSS
LL  II      SS
LL  II      SS
LL  II      SS
LL  II      SS
LLLLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLLLL  IIIIII  SSSSSSSS
```


| | | |
|-----|------|---|
| (1) | 695 | CR11 FUNCTION DECISION TABLE |
| (1) | 733 | CANCEL I/O ON CHANNEL |
| (1) | 761 | READ FUNCTION PROCESSING |
| (1) | 833 | START I/O OPERATION ON CR11 CARD READER |
| (1) | 1016 | CR11 CARD READER INTDERRUPTS |
| (1) | 1072 | CARD READER INITIALIZATION |
| (1) | 1094 | CARD READER UNIT INITIALIZATION |

```
0000 1 .TITLE CRDRIVER - CR11 CARD READER DRIVER
0000 2 .IDENT 'V04-000'
0000 3
0000 4
0000 5 *****
0000 6 *
0000 7 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 8 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 9 * ALL RIGHTS RESERVED.
0000 10 *
0000 11 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 12 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 13 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 14 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 15 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 16 * TRANSFERRED.
0000 17 *
0000 18 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 19 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 20 * CORPORATION.
0000 21 *
0000 22 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 23 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 24 *
0000 25 *
0000 26 *****
0000 27
0000 28 D. N. CUTLER 1-SEP-77
0000 29
0000 30 MODIFICATION HISTORY:
0000 31
0000 32 V03-002 EMD0087 Ellen M. Dusseault 30-Apr-1984
0000 33 Add DEV$M_NNM characteristic to DEVCHAR2 so that these
0000 34 devices will have the 'node$' prefix.
0000 35
0000 36 V03-001 KDM0002 Kathleen D. Morse 28-Jun-1982
0000 37 Added $DYNDEF, $SSDEF, $DCDEF, and $PRDEF.
0000 38
0000 39
0000 40 MACRO LIBRARY CALLS
0000 41
0000 42
0000 43 $CRBDEF ;DEFINE CRB OFFSETS
0000 44 $CRDEF ;DEFINE CARD READER STATUS BITS
0000 45 $DCDEF ;DEFINE ADAPTER TYPES
0000 46 $DDBDEF ;DEFINE DDB OFFSETS
0000 47 $DPTDEF ;DEFINE DPT OFFSETS
0000 48 $DYNDEF ;DEFINE DYNAMIC DATA STRUCTURE TYPES
0000 49 $IDBDEF ;DEFINE IDB OFFSETS
0000 50 $IODEF ;DEFINE I/O FUNCTION CODES
0000 51 $IRPDEF ;DEFINE IRP OFFSETS
0000 52 $JIBDEF ;DEFINE JIB OFFSETS
0000 53 $MSGDEF ;DEFINE SYSTEM MESSAGE TYPES
0000 54 $PCBDEF ;DEFINE PCB OFFSETS
0000 55 $PRDEF ;DEFINE PROCESSOR REGISTERS
0000 56 $UCBDEF ;DEFINE UCB OFFSETS
0000 57 $SSDEF ;DEFINE STATUS CODES
```



```
0000 58          SVECDEF                      ;DEFINE VEC OFFSETS
0000 59
0000 60 :
0000 61 : LOCAL SYMBOLS
0000 62 :
0000 63 : ARGUMENT LIST OFFSET DEFINITIONS
0000 64 :
0000 65
00000000 0000 66 P1=0                      ;FIRST FUNCTION DEPENDENT PARAMETER
00000004 0000 67 P2=4                      ;SECOND FUNCTION DEPENDENT PARAMETER
00000008 0000 68 P3=8                      ;THIRD FUNCTION DEPENDENT PARAMETER
0000000C 0000 69 P4=12                     ;FOURTH FUNCTION DEPENDENT PARAMETER
00000010 0000 70 P5=16                     ;FIFTH FUNCTION DEPENDENT PARAMETER
00000014 0000 71 P6=20                     ;SIZTH FUNCTION DEPENDENT PARAMETER
0000 72
0000 73 :
0000 74 : SPECIAL CARD COLUMN PATTERNS
0000 75 :
0000 76
00000F0F 0000 77 CR_EOF="B111100001111    ;END OF FILE (12-11-0-1-6-7-8-9)
000008A2 0000 78 CR_026="B100010100010    ;TRANSLATE 026 CARD CODE (12-2-4-8)
00000AAA 0000 79 CR_029="B101010101010    ;TRANSLATE 029 CARD CODE (12-0-2-4-6-8)
0000 80
0000 81 :
0000 82 : CR11 CONTROLLER REGISTER OFFSET DEFINITIONS
0000 83 :
0000 84
0000 85          $DEFINI CR
0000 86
0000 87 $DEF    CR_CSR      .BLKW    1      ;CONTROL STATUS REGISTER
0002 88      _VIED    CR_CSR,0,<-          ;CONTROL STATUS REGISTER FIELD DEFINITIONS
0002 89              <READ,,M>,-          ;READ CARD
0002 90              <EJECT,,M>,-          ;EJECT CARD
0002 91              <,4>,-                ;RESERVED BITS
0002 92              <IE,,M>,-             ;INTERRUPT ENABLE
0002 93              <CLDONE,,M>,-          ;COLUMN DONE
0002 94              <OFFLIN,,M>,-          ;READER OFFLINE
0002 95              <BUSY,,M>,-            ;CARD BEING READ
0002 96              <ONLINE,,M>,-          ;READER ONLINE
0002 97              <TIMERR,,M>,-          ;TIMING ERROR
0002 98              <MCHECK,,M>,-          ;MOTION CHECK
0002 99              <HCHECK,,M>,-          ;HOPPER CHECK
0002 100             <CRDONE,,M>,-          ;CARD DONE
0002 101             <ERROR,,M>,-          ;ERROR CONDITION
0002 102             >
0002 103 $DEF    CR_CRB1      .BLKW    1      ;CARD READ DATA BUFFER 1 (BINARY)
0004 104 $DEF    CR_CRB2      .BLKW    1      ;CARD READ DATA BUFFER 2 (PACKED)
0006 105
0006 106          $DEFEND CR
0000 107
0000 108 :
0000 109 : DEFINE DEVICE DEPENDENT UNIT CONTROL BLOCK OFFSETS
0000 110 :
0000 111
0000 112          $DEFINI UCB
0000 113
00000090 0000 114 .=UCB$K_LENGTH          ;
```



```
0090 115
0090 116 $DEF UCBSB_CR_COLCNT .BLKB 1 ;CURRENT COLUMN COUNT
0091 117 $DEF UCBSB_CR_EOFcnt .BLKB 1 ;END OF FILE PUNCH COUNT
0092 118 $DEF UCBSB_CR_EOFcol .BLKB 1 ;NUMBER OF END OF FILE PUNCHES REQUIRED
0093 119 $DEF UCBSB_CR_OFcnt .BLKB 1 ;OFFLINE TIME COUNTER
0094 120 $DEF UCBSW_CR_FSTCOL .BLKW 1 ;FIRST COLUMN BINARY DATA
0096 121 $DEF UCBSW_CR_CSR .BLKW 1 ;SAVED FINAL CONTROL STATUS REGISTER
0098 122
00000098 0098 123 UCBSK_CR_LENGTH=.
0098 124
0098 125 $DEFEND UCB
0000 126
0000 127 :
0000 128 : LOCAL DATA
0000 129 :
0000 130 : DRIVER PROLOGUE TABLE
0000 131 :
0000 132
0000 133 DPTAB - ;DEFINE DRIVER PROLOGUE TABLE
0000 134 END=CR_END,- ;END OF DRIVER
0000 135 ADAPTER=UBA,- ;ADAPTER TYPE
0000 136 UCBSIZE=UCBSK_CR_LENGTH,- ;UCB SIZE
0000 137 NAME=CRDRIVER- ;DRIVER NAME
0038 138 DPT_STORE INIT ;CONTROL BLOCK INIT VALUES
0038 139 DPT_STORE UCB,UCBSB_FIPL,B,8 ;FORK IPL
003C 140 DPT_STORE UCB,UCBSL_DEVCHAR,L,- ;DEVICE CHARACTERISTICS
003C 141 <DEVSM_REC- ;RECORD ORIENTED
003C 142 !DEVSM_AVL- ;AVAILABLE
003C 143 !DEVSM_IDV> ;INPUT DEVICE
0043 144 DPT_STORE UCB,UCBSL_DEVCHAR2,L,- ;DEVICE CHARACTERISTIC
0043 145 <DEVSM_NNM> ;PREFIX WITH 'NODES'
004A 146 DPT_STORE UCB,UCBSB_DEVCLASS,B,DC$ CARD ;DEVICE CLASS
004E 147 DPT_STORE UCB,UCBSB_DEVTYPE,B,DT$ CR11 ;DEVICE TYPE
0052 148 DPT_STORE UCB,UCBSW_DEVBUFSIZ,W,80 ;DEFAULT BUFFER SIZE
0057 149 DPT_STORE UCB,UCBSL_DEVDEPEND,L,CR$K TO29 ;DEFAULT TRANSLATION MODE
005E 150 DPT_STORE UCB,UCBSB_DIPL,B,22 ;DEVICE IPL
0062 151 DPT_STORE REINIT ;CONTROL BLOCK RE-INIT VALUES
0062 152 DPT_STORE CRB,CRBSL_INTD+4,D,CR$INT ;INTERRUPT SERVICE ROUTINE ADDRESS
0067 153 DPT_STORE CRB,CRBSL_INTD+VEC$ INITIAL,D,CR ;CONTROLLER INIT
006C 154 DPT_STORE CRB,CRBSL_INTD+VEC$ UNITINIT,D,CR CR11_INIT ;UNIT INIT
0071 155 DPT_STORE DDB,DDBSL_DDT,D,CR$DDT ;DDT ADDRESS
0076 156 DPT_STORE END ;
0000 157
0000 158 :
0000 159 : DRIVER DISPATCH TABLE
0000 160 :
0000 161
0000 162 DDTAB CR,- ;DRIVER DISPATCH TABLE
0000 163 CR_STARTIO,- ;START I/O OPERATION
0000 164 0,- ;UNSOLICITED INTERRUPT
0000 165 CR_FUNCtable,- ;FUNCTION DECISION TABLE
0000 166 CR_CANCELIO,- ;CANCEL I/O OPERATION
0000 167 0,- ;REGISTER DUMP ROUTINE
0000 168 0,- ;SIZE OF DIAGNOSTIC BUFFER
0000 169 0 ;SIZE OF ERROR LOG BUFFER
0038 170
0038 171 :
```



```
0038 172 : 029 CONVERSION TABLE
0038 173 :
0038 174 :
0038 175 CR_CVT029:
20 0038 176 .BYTE ^A/ /
31 0039 177 .BYTE ^A/1/
32 003A 178 .BYTE ^A/2/
33 003B 179 .BYTE ^A/3/
34 003C 180 .BYTE ^A/4/
35 003D 181 .BYTE ^A/5/
36 003E 182 .BYTE ^A/6/
37 003F 183 .BYTE ^A/7/
38 0040 184 .BYTE ^A/8/
60 0041 185 .BYTE ^A/ /
3A 0042 186 .BYTE ^A/:/
23 0043 187 .BYTE ^A/#/
40 0044 188 .BYTE ^A/@/
27 0045 189 .BYTE ^A/'/
3D 0046 190 .BYTE ^A/=/
22 0047 191 .BYTE ^A/'/'
39 0048 192 .BYTE ^A/9/
5C 0049 193 .BYTE ^A/\ /
16 004A 194 .BYTE ^X16
5C 004B 195 .BYTE ^A/\ /
5C 004C 196 .BYTE ^A/\ /
5C 004D 197 .BYTE ^A/\ /
5C 004E 198 .BYTE ^A/\ /
04 004F 199 .BYTE ^X04
5C 0050 200 .BYTE ^A/\ /
5C 0051 201 .BYTE ^A/\ /
5C 0052 202 .BYTE ^A/\ /
5C 0053 203 .BYTE ^A/\ /
14 0054 204 .BYTE ^X14
15 0055 205 .BYTE ^X15
5C 0056 206 .BYTE ^A/\ /
1A 0057 207 .BYTE ^X1A
30 0058 208 .BYTE ^A/0/
2F 0059 209 .BYTE ^A/\ /
53 005A 210 .BYTE ^A/S/
54 005B 211 .BYTE ^A/T/
55 005C 212 .BYTE ^A/U/
56 005D 213 .BYTE ^A/V/
57 005E 214 .BYTE ^A/W/
58 005F 215 .BYTE ^A/X/
59 0060 216 .BYTE ^A/Y/
5C 0061 217 .BYTE ^A/\ /
5C 0062 218 .BYTE ^A/\ /
2C 0063 219 .BYTE ^A/ /
25 0064 220 .BYTE ^A/ /
5F 0065 221 .BYTE ^A/ /
3E 0066 222 .BYTE ^A/S/
3F 0067 223 .BYTE ^A/?/
5A 0068 224 .BYTE ^A/Z/
5C 0069 225 .BYTE ^A/\ /
5C 006A 226 .BYTE ^A/\ /
5C 006B 227 .BYTE ^A/\ /
5C 006C 228 .BYTE ^A/\ /
```

```
:029 TRANSLATE TABLE
0 ^X00 ^0000
1 ^X01 ^0001
2 ^X02 ^0002
3 ^X03 ^0003
4 ^X04 ^0004
5 ^X05 ^0005
6 ^X06 ^0006
7 ^X07 ^0007
8 ^X08 ^0010
9 ^X09 ^0011
10 ^X0A ^0012
11 ^X0B ^0013
12 ^X0C ^0014
13 ^X0D ^0015
14 ^X0E ^0016
15 ^X0F ^0017
16 ^X10 ^0020
17 ^X11 ^0021
18 ^X12 ^0022
19 ^X13 ^0023
20 ^X14 ^0024
21 ^X15 ^0025
22 ^X16 ^0026
23 ^X17 ^0027
24 ^X18 ^0030
25 ^X19 ^0031
26 ^X1A ^0032
27 ^X1B ^0033
28 ^X1C ^0034
29 ^X1D ^0035
30 ^X1E ^0036
31 ^X1F ^0037
32 ^X20 ^0040
33 ^X21 ^0041
34 ^X22 ^0042
35 ^X23 ^0043
36 ^X24 ^0044
37 ^X25 ^0045
38 ^X26 ^0046
39 ^X27 ^0047
40 ^X28 ^0050
41 ^X29 ^0051
42 ^X2A ^0052
43 ^X2B ^0053
44 ^X2C ^0054
45 ^X2D ^0055
46 ^X2E ^0056
47 ^X2F ^0057
48 ^X30 ^0060
49 ^X31 ^0061
50 ^X32 ^0062
51 ^X33 ^0063
52 ^X34 ^0064
```


| | | | | | | | |
|----|------|-----|-------|-------|-----|------|-------|
| 0A | 006D | 229 | .BYTE | ^X0A | 53 | ^X35 | ^0065 |
| 17 | 006E | 230 | .BYTE | ^X17 | 54 | ^X36 | ^0066 |
| 1B | 006F | 231 | .BYTE | ^X1B | 55 | ^X37 | ^0067 |
| 5C | 0070 | 232 | .BYTE | ^A/\ | 56 | ^X38 | ^0070 |
| 5C | 0071 | 233 | .BYTE | ^A/\ | 57 | ^X39 | ^0071 |
| 5C | 0072 | 234 | .BYTE | ^A/\ | 58 | ^X3A | ^0072 |
| 5C | 0073 | 235 | .BYTE | ^A/\ | 59 | ^X3B | ^0073 |
| 5C | 0074 | 236 | .BYTE | ^A/\ | 60 | ^X3C | ^0074 |
| 05 | 0075 | 237 | .BYTE | ^X05 | 61 | ^X3D | ^0075 |
| 06 | 0076 | 238 | .BYTE | ^X06 | 62 | ^X3E | ^0076 |
| 07 | 0077 | 239 | .BYTE | ^X07 | 63 | ^X3F | ^0077 |
| 2D | 0078 | 240 | .BYTE | ^A/- | 64 | ^X40 | ^0100 |
| 4A | 0079 | 241 | .BYTE | ^A/J | 65 | ^X41 | ^0101 |
| 4B | 007A | 242 | .BYTE | ^A/K | 66 | ^X42 | ^0102 |
| 4C | 007B | 243 | .BYTE | ^A/L | 67 | ^X43 | ^0103 |
| 4D | 007C | 244 | .BYTE | ^A/M | 68 | ^X44 | ^0104 |
| 4E | 007D | 245 | .BYTE | ^A/N | 69 | ^X45 | ^0105 |
| 4F | 007E | 246 | .BYTE | ^A/O | 70 | ^X46 | ^0106 |
| 50 | 007F | 247 | .BYTE | ^A/P | 71 | ^X47 | ^0107 |
| 51 | 0080 | 248 | .BYTE | ^A/Q | 72 | ^X48 | ^0110 |
| 5C | 0081 | 249 | .BYTE | ^A/\ | 73 | ^X49 | ^0111 |
| 5D | 0082 | 250 | .BYTE | ^A/J | 74 | ^X4A | ^0112 |
| 24 | 0083 | 251 | .BYTE | ^A/\$ | 75 | ^X4B | ^0113 |
| 2A | 0084 | 252 | .BYTE | ^A/* | 76 | ^X4C | ^0114 |
| 29 | 0085 | 253 | .BYTE | ^A/) | 77 | ^X4D | ^0115 |
| 3B | 0086 | 254 | .BYTE | ^A/: | 78 | ^X4E | ^0116 |
| 5E | 0087 | 255 | .BYTE | ^A/^ | 79 | ^X4F | ^0117 |
| 52 | 0088 | 256 | .BYTE | ^A/R | 80 | ^X50 | ^0120 |
| 11 | 0089 | 257 | .BYTE | ^X11 | 81 | ^X51 | ^0121 |
| 12 | 008A | 258 | .BYTE | ^X12 | 82 | ^X52 | ^0122 |
| 13 | 008B | 259 | .BYTE | ^X13 | 83 | ^X53 | ^0123 |
| 5C | 008C | 260 | .BYTE | ^A/\ | 84 | ^X54 | ^0124 |
| 5C | 008D | 261 | .BYTE | ^A/\ | 85 | ^X55 | ^0125 |
| 08 | 008E | 262 | .BYTE | ^X08 | 86 | ^X56 | ^0126 |
| 5C | 008F | 263 | .BYTE | ^A/\ | 87 | ^X57 | ^0127 |
| 18 | 0090 | 264 | .BYTE | ^X18 | 88 | ^X58 | ^0130 |
| 19 | 0091 | 265 | .BYTE | ^X19 | 89 | ^X59 | ^0131 |
| 5C | 0092 | 266 | .BYTE | ^A/\ | 90 | ^X5A | ^0132 |
| 5C | 0093 | 267 | .BYTE | ^A/\ | 91 | ^X5B | ^0133 |
| 1C | 0094 | 268 | .BYTE | ^X1C | 92 | ^X5C | ^0134 |
| 1D | 0095 | 269 | .BYTE | ^X1D | 93 | ^X5D | ^0135 |
| 1E | 0096 | 270 | .BYTE | ^X1E | 94 | ^X5E | ^0136 |
| 1F | 0097 | 271 | .BYTE | ^X1F | 95 | ^X5F | ^0137 |
| 7D | 0098 | 272 | .BYTE | ^A/) | 96 | ^X60 | ^0140 |
| 7E | 0099 | 273 | .BYTE | ^A/- | 97 | ^X61 | ^0141 |
| 73 | 009A | 274 | .BYTE | ^A/s | 98 | ^X62 | ^0142 |
| 74 | 009B | 275 | .BYTE | ^A/t | 99 | ^X63 | ^0143 |
| 75 | 009C | 276 | .BYTE | ^A/u | 100 | ^X64 | ^0144 |
| 76 | 009D | 277 | .BYTE | ^A/v | 101 | ^X65 | ^0145 |
| 77 | 009E | 278 | .BYTE | ^A/w | 102 | ^X66 | ^0146 |
| 78 | 009F | 279 | .BYTE | ^A/x | 103 | ^X67 | ^0147 |
| 79 | 00A0 | 280 | .BYTE | ^A/y | 104 | ^X68 | ^0150 |
| 5C | 00A1 | 281 | .BYTE | ^A/\ | 105 | ^X69 | ^0151 |
| 5C | 00A2 | 282 | .BYTE | ^A/\ | 106 | ^X6A | ^0152 |
| 5C | 00A3 | 283 | .BYTE | ^A/\ | 107 | ^X6B | ^0153 |
| 5C | 00A4 | 284 | .BYTE | ^A/\ | 108 | ^X6C | ^0154 |
| 5C | 00A5 | 285 | .BYTE | ^A/\ | 109 | ^X6D | ^0155 |

| | | | | | | | | |
|----|------|-----|-------|------|---|-----|------|-------|
| 5C | 00A6 | 286 | .BYTE | ^A/\ | : | 110 | ^X6E | ^0156 |
| 5C | 00A7 | 287 | .BYTE | ^A/\ | : | 111 | ^X6F | ^0157 |
| 7A | 00A8 | 288 | .BYTE | ^A/z | : | 112 | ^X70 | ^0160 |
| 5C | 00A9 | 289 | .BYTE | ^A/\ | : | 113 | ^X71 | ^0161 |
| 5C | 00AA | 290 | .BYTE | ^A/\ | : | 114 | ^X72 | ^0162 |
| 5C | 00AB | 291 | .BYTE | ^A/\ | : | 115 | ^X73 | ^0163 |
| 5C | 00AC | 292 | .BYTE | ^A/\ | : | 116 | ^X74 | ^0164 |
| 5C | 00AD | 293 | .BYTE | ^A/\ | : | 117 | ^X75 | ^0165 |
| 5C | 00AE | 294 | .BYTE | ^A/\ | : | 118 | ^X76 | ^0166 |
| 5C | 00AF | 295 | .BYTE | ^A/\ | : | 119 | ^X77 | ^0167 |
| 5C | 00B0 | 296 | .BYTE | ^A/\ | : | 120 | ^X78 | ^0170 |
| 5C | 00B1 | 297 | .BYTE | ^A/\ | : | 121 | ^X79 | ^0171 |
| 5C | 00B2 | 298 | .BYTE | ^A/\ | : | 122 | ^X7A | ^0172 |
| 5C | 00B3 | 299 | .BYTE | ^A/\ | : | 123 | ^X7B | ^0173 |
| 5C | 00B4 | 300 | .BYTE | ^A/\ | : | 124 | ^X7C | ^0174 |
| 5C | 00B5 | 301 | .BYTE | ^A/\ | : | 125 | ^X7D | ^0175 |
| 5C | 00B6 | 302 | .BYTE | ^A/\ | : | 126 | ^X7E | ^0176 |
| 5C | 00B7 | 303 | .BYTE | ^A/\ | : | 127 | ^X7F | ^0177 |
| 26 | 00B8 | 304 | .BYTE | ^A/& | : | 128 | ^X80 | ^0200 |
| 41 | 00B9 | 305 | .BYTE | ^A/A | : | 129 | ^X81 | ^0201 |
| 42 | 00BA | 306 | .BYTE | ^A/B | : | 130 | ^X82 | ^0202 |
| 43 | 00BB | 307 | .BYTE | ^A/C | : | 131 | ^X83 | ^0203 |
| 44 | 00BC | 308 | .BYTE | ^A/D | : | 132 | ^X84 | ^0204 |
| 45 | 00BD | 309 | .BYTE | ^A/E | : | 133 | ^X85 | ^0205 |
| 46 | 00BE | 310 | .BYTE | ^A/F | : | 134 | ^X86 | ^0206 |
| 47 | 00BF | 311 | .BYTE | ^A/G | : | 135 | ^X87 | ^0207 |
| 48 | 00C0 | 312 | .BYTE | ^A/H | : | 136 | ^X88 | ^0210 |
| 5C | 00C1 | 313 | .BYTE | ^A/\ | : | 137 | ^X89 | ^0211 |
| 5B | 00C2 | 314 | .BYTE | ^A/C | : | 138 | ^X8A | ^0212 |
| 2E | 00C3 | 315 | .BYTE | ^A/. | : | 139 | ^X8B | ^0213 |
| 3C | 00C4 | 316 | .BYTE | ^A/< | : | 140 | ^X8C | ^0214 |
| 28 | 00C5 | 317 | .BYTE | ^A/C | : | 141 | ^X8D | ^0215 |
| 2B | 00C6 | 318 | .BYTE | ^A/+ | : | 142 | ^X8E | ^0216 |
| 21 | 00C7 | 319 | .BYTE | ^A/! | : | 143 | ^X8F | ^0217 |
| 49 | 00C8 | 320 | .BYTE | ^A/I | : | 144 | ^X90 | ^0220 |
| 01 | 00C9 | 321 | .BYTE | ^X01 | : | 145 | ^X91 | ^0221 |
| 02 | 00CA | 322 | .BYTE | ^X02 | : | 146 | ^X92 | ^0222 |
| 03 | 00CB | 323 | .BYTE | ^X03 | : | 147 | ^X93 | ^0223 |
| 5C | 00CC | 324 | .BYTE | ^A/\ | : | 148 | ^X94 | ^0224 |
| 09 | 00CD | 325 | .BYTE | ^X09 | : | 149 | ^X95 | ^0225 |
| 5C | 00CE | 326 | .BYTE | ^A/\ | : | 150 | ^X96 | ^0226 |
| FF | 00CF | 327 | .BYTE | ^XFF | : | 151 | ^X97 | ^0227 |
| 5C | 00D0 | 328 | .BYTE | ^A/\ | : | 152 | ^X98 | ^0230 |
| 5C | 00D1 | 329 | .BYTE | ^A/\ | : | 153 | ^X99 | ^0231 |
| 5C | 00D2 | 330 | .BYTE | ^A/\ | : | 154 | ^X9A | ^0232 |
| 0B | 00D3 | 331 | .BYTE | ^X0B | : | 155 | ^X9B | ^0233 |
| 0C | 00D4 | 332 | .BYTE | ^X0C | : | 156 | ^X9C | ^0234 |
| 0D | 00D5 | 333 | .BYTE | ^X0D | : | 157 | ^X9D | ^0235 |
| 0E | 00D6 | 334 | .BYTE | ^X0E | : | 158 | ^X9E | ^0236 |
| 0F | 00D7 | 335 | .BYTE | ^X0F | : | 159 | ^X9F | ^0237 |
| 7B | 00D8 | 336 | .BYTE | ^A/C | : | 160 | ^XA0 | ^0240 |
| 61 | 00D9 | 337 | .BYTE | ^A/a | : | 161 | ^XA1 | ^0241 |
| 62 | 00DA | 338 | .BYTE | ^A/b | : | 162 | ^XA2 | ^0242 |
| 63 | 00DB | 339 | .BYTE | ^A/c | : | 163 | ^XA3 | ^0243 |
| 64 | 00DC | 340 | .BYTE | ^A/d | : | 164 | ^XA4 | ^0244 |
| 65 | 00DD | 341 | .BYTE | ^A/e | : | 165 | ^XA5 | ^0245 |
| 66 | 00DE | 342 | .BYTE | ^A/f | : | 166 | ^XA6 | ^0246 |

| | | | | | | | |
|----|------|-----|-------|-------|-----|------|-------|
| 67 | 00DF | 343 | .BYTE | *A/g/ | 167 | *XA7 | *0247 |
| 68 | 00E0 | 344 | .BYTE | *A/h/ | 168 | *XA8 | *0250 |
| 5C | 00E1 | 345 | .BYTE | *A/i/ | 169 | *XA9 | *0251 |
| 5C | 00E2 | 346 | .BYTE | *A/j/ | 170 | *XAA | *0252 |
| 5C | 00E3 | 347 | .BYTE | *A/k/ | 171 | *XAB | *0253 |
| 5C | 00E4 | 348 | .BYTE | *A/l/ | 172 | *XAC | *0254 |
| 5C | 00E5 | 349 | .BYTE | *A/m/ | 173 | *XAD | *0255 |
| 5C | 00E6 | 350 | .BYTE | *A/n/ | 174 | *XAE | *0256 |
| 5C | 00E7 | 351 | .BYTE | *A/o/ | 175 | *XAF | *0257 |
| 69 | 00E8 | 352 | .BYTE | *A/p/ | 176 | *XB0 | *0260 |
| 5C | 00E9 | 353 | .BYTE | *A/q/ | 177 | *XB1 | *0261 |
| 5C | 00EA | 354 | .BYTE | *A/r/ | 178 | *XB2 | *0262 |
| 5C | 00EB | 355 | .BYTE | *A/s/ | 179 | *XB3 | *0263 |
| 5C | 00EC | 356 | .BYTE | *A/t/ | 180 | *XB4 | *0264 |
| 5C | 00ED | 357 | .BYTE | *A/u/ | 181 | *XB5 | *0265 |
| 5C | 00EE | 358 | .BYTE | *A/v/ | 182 | *XB6 | *0266 |
| 5C | 00EF | 359 | .BYTE | *A/w/ | 183 | *XB7 | *0267 |
| 5C | 00F0 | 360 | .BYTE | *A/x/ | 184 | *XB8 | *0270 |
| 5C | 00F1 | 361 | .BYTE | *A/y/ | 185 | *XB9 | *0271 |
| 5C | 00F2 | 362 | .BYTE | *A/z/ | 186 | *XBA | *0272 |
| 5C | 00F3 | 363 | .BYTE | *A/0/ | 187 | *XBB | *0273 |
| 5C | 00F4 | 364 | .BYTE | *A/1/ | 188 | *XBC | *0274 |
| 5C | 00F5 | 365 | .BYTE | *A/2/ | 189 | *XBD | *0275 |
| 5C | 00F6 | 366 | .BYTE | *A/3/ | 190 | *XBE | *0276 |
| 5C | 00F7 | 367 | .BYTE | *A/4/ | 191 | *XBF | *0277 |
| 7C | 00F8 | 368 | .BYTE | *A/5/ | 192 | *XC0 | *0300 |
| 6A | 00F9 | 369 | .BYTE | *A/6/ | 193 | *XC1 | *0301 |
| 6B | 00FA | 370 | .BYTE | *A/7/ | 194 | *XC2 | *0302 |
| 6C | 00FB | 371 | .BYTE | *A/8/ | 195 | *XC3 | *0303 |
| 6D | 00FC | 372 | .BYTE | *A/9/ | 196 | *XC4 | *0304 |
| 6E | 00FD | 373 | .BYTE | *A/A/ | 197 | *XC5 | *0305 |
| 6F | 00FE | 374 | .BYTE | *A/B/ | 198 | *XC6 | *0306 |
| 70 | 00FF | 375 | .BYTE | *A/C/ | 199 | *XC7 | *0307 |
| 71 | 0100 | 376 | .BYTE | *A/D/ | 200 | *XC8 | *0310 |
| 5C | 0101 | 377 | .BYTE | *A/E/ | 201 | *XC9 | *0311 |
| 5C | 0102 | 378 | .BYTE | *A/F/ | 202 | *XCA | *0312 |
| 5C | 0103 | 379 | .BYTE | *A/G/ | 203 | *XCB | *0313 |
| 5C | 0104 | 380 | .BYTE | *A/H/ | 204 | *XCC | *0314 |
| 5C | 0105 | 381 | .BYTE | *A/I/ | 205 | *XCD | *0315 |
| 5C | 0106 | 382 | .BYTE | *A/J/ | 206 | *XCE | *0316 |
| 5C | 0107 | 383 | .BYTE | *A/K/ | 207 | *XCF | *0317 |
| 72 | 0108 | 384 | .BYTE | *A/L/ | 208 | *XD0 | *0320 |
| 5C | 0109 | 385 | .BYTE | *A/M/ | 209 | *XD1 | *0321 |
| 5C | 010A | 386 | .BYTE | *A/N/ | 210 | *XD2 | *0322 |
| 5C | 010B | 387 | .BYTE | *A/O/ | 211 | *XD3 | *0323 |
| 5C | 010C | 388 | .BYTE | *A/P/ | 212 | *XD4 | *0324 |
| 5C | 010D | 389 | .BYTE | *A/Q/ | 213 | *XD5 | *0325 |
| 5C | 010E | 390 | .BYTE | *A/R/ | 214 | *XD6 | *0326 |
| 5C | 010F | 391 | .BYTE | *A/S/ | 215 | *XD7 | *0327 |
| 5C | 0110 | 392 | .BYTE | *A/T/ | 216 | *XD8 | *0330 |
| 10 | 0111 | 393 | .BYTE | *A/U/ | 217 | *XD9 | *0331 |
| 5C | 0112 | 394 | .BYTE | *A/V/ | 218 | *XDA | *0332 |
| 5C | 0113 | 395 | .BYTE | *A/W/ | 219 | *XDB | *0333 |
| 5C | 0114 | 396 | .BYTE | *A/X/ | 220 | *XDC | *0334 |
| 5C | 0115 | 397 | .BYTE | *A/Y/ | 221 | *XDD | *0335 |
| 5C | 0116 | 398 | .BYTE | *A/Z/ | 222 | *XDE | *0336 |
| 5C | 0117 | 399 | .BYTE | *A/0/ | 223 | *XDF | *0337 |

| | | | | | | | | |
|----|------|-----|-------|-------|---|-----|------|-------|
| 5C | 0118 | 400 | .BYTE | ^A/\ | : | 224 | ^XE0 | ^0340 |
| 5C | 0119 | 401 | .BYTE | ^A/\ | : | 225 | ^XE1 | ^0341 |
| 5C | 011A | 402 | .BYTE | ^A/\ | : | 226 | ^XE2 | ^0342 |
| 5C | 011B | 403 | .BYTE | ^A/\ | : | 227 | ^XE3 | ^0343 |
| 5C | 011C | 404 | .BYTE | ^A/\ | : | 228 | ^XE4 | ^0344 |
| 5C | 011D | 405 | .BYTE | ^A/\ | : | 229 | ^XE5 | ^0345 |
| 5C | 011E | 406 | .BYTE | ^A/\ | : | 230 | ^XE6 | ^0346 |
| 5C | 011F | 407 | .BYTE | ^A/\ | : | 231 | ^XE7 | ^0347 |
| 5C | 0120 | 408 | .BYTE | ^A/\ | : | 232 | ^XE8 | ^0350 |
| 5C | 0121 | 409 | .BYTE | ^A/\ | : | 233 | ^XE9 | ^0351 |
| 5C | 0122 | 410 | .BYTE | ^A/\ | : | 234 | ^XEA | ^0352 |
| 5C | 0123 | 411 | .BYTE | ^A/\ | : | 235 | ^XEB | ^0353 |
| 5C | 0124 | 412 | .BYTE | ^A/\ | : | 236 | ^XEC | ^0354 |
| 5C | 0125 | 413 | .BYTE | ^A/\ | : | 237 | ^XED | ^0355 |
| 5C | 0126 | 414 | .BYTE | ^A/\ | : | 238 | ^XEE | ^0356 |
| 5C | 0127 | 415 | .BYTE | ^A/\ | : | 239 | ^XEF | ^0357 |
| 5C | 0128 | 416 | .BYTE | ^A/\ | : | 240 | ^XF0 | ^0360 |
| 5C | 0129 | 417 | .BYTE | ^A/\ | : | 241 | ^XF1 | ^0361 |
| 5C | 012A | 418 | .BYTE | ^A/\ | : | 242 | ^XF2 | ^0362 |
| 5C | 012B | 419 | .BYTE | ^A/\ | : | 243 | ^XF3 | ^0363 |
| 5C | 012C | 420 | .BYTE | ^A/\ | : | 244 | ^XF4 | ^0364 |
| 5C | 012D | 421 | .BYTE | ^A/\ | : | 245 | ^XF5 | ^0365 |
| 5C | 012E | 422 | .BYTE | ^A/\ | : | 246 | ^XF6 | ^0366 |
| 5C | 012F | 423 | .BYTE | ^A/\ | : | 247 | ^XF7 | ^0367 |
| 5C | 0130 | 424 | .BYTE | ^A/\ | : | 248 | ^XF8 | ^0370 |
| 5C | 0131 | 425 | .BYTE | ^A/\ | : | 249 | ^XF9 | ^0371 |
| 5C | 0132 | 426 | .BYTE | ^A/\ | : | 250 | ^XFA | ^0372 |
| 5C | 0133 | 427 | .BYTE | ^A/\ | : | 251 | ^XFB | ^0373 |
| 5C | 0134 | 428 | .BYTE | ^A/\ | : | 252 | ^XFC | ^0374 |
| 5C | 0135 | 429 | .BYTE | ^A/\ | : | 253 | ^XFD | ^0375 |
| 5C | 0136 | 430 | .BYTE | ^A/\ | : | 254 | ^XFE | ^0376 |
| 5C | 0137 | 431 | .BYTE | ^A/\ | : | 255 | ^XFF | ^0377 |
| | 0138 | 432 | | | | | | |
| | 0138 | 433 | | | | | | |
| | 0138 | 434 | | | | | | |
| | 0138 | 435 | | | | | | |
| | 0138 | 436 | | | | | | |
| | 0138 | 437 | | | | | | |
| | 0138 | 438 | | | | | | |
| 20 | 0139 | 439 | .BYTE | ^A/ / | : | 0 | ^X00 | ^0000 |
| 31 | 013A | 440 | .BYTE | ^A/1/ | : | 1 | ^X01 | ^0001 |
| 32 | 013B | 441 | .BYTE | ^A/2/ | : | 2 | ^X02 | ^0002 |
| 33 | 013C | 442 | .BYTE | ^A/3/ | : | 3 | ^X03 | ^0003 |
| 34 | 013D | 443 | .BYTE | ^A/4/ | : | 4 | ^X04 | ^0004 |
| 35 | 013E | 444 | .BYTE | ^A/5/ | : | 5 | ^X05 | ^0005 |
| 36 | 013F | 445 | .BYTE | ^A/6/ | : | 6 | ^X06 | ^0006 |
| 37 | 0140 | 446 | .BYTE | ^A/7/ | : | 7 | ^X07 | ^0007 |
| 38 | 0141 | 447 | .BYTE | ^A/8/ | : | 8 | ^X08 | ^0010 |
| 60 | 0142 | 448 | .BYTE | ^A/ / | : | 9 | ^X09 | ^0011 |
| 5F | 0143 | 449 | .BYTE | ^A/ / | : | 10 | ^X0A | ^0012 |
| 3D | 0144 | 450 | .BYTE | ^A/=/ | : | 11 | ^X0B | ^0013 |
| 40 | 0145 | 451 | .BYTE | ^A/2/ | : | 12 | ^X0C | ^0014 |
| 5E | 0146 | 452 | .BYTE | ^A/ / | : | 13 | ^X0D | ^0015 |
| 27 | 0147 | 453 | .BYTE | ^A/ / | : | 14 | ^X0E | ^0016 |
| 5C | 0148 | 454 | .BYTE | ^A/\ | : | 15 | ^X0F | ^0017 |
| 39 | 0149 | 455 | .BYTE | ^A/9/ | : | 16 | ^X10 | ^0020 |
| 5C | 014A | 456 | .BYTE | ^A/\ | : | 17 | ^X11 | ^0021 |
| 16 | | | .BYTE | ^X16 | : | 18 | ^X12 | ^0022 |

026 CONVERSION TABLE

CR_CVT026:

026 TRANSLATE TABLE

| | | | | | | | |
|----|------|-----|-------|------|----|------|-------|
| 5C | 014B | 457 | .BYTE | ^A/\ | 19 | ^X13 | ^0023 |
| 5C | 014C | 458 | .BYTE | ^A/\ | 20 | ^X14 | ^0024 |
| 5C | 014D | 459 | .BYTE | ^A/\ | 21 | ^X15 | ^0025 |
| 5C | 014E | 460 | .BYTE | ^A/\ | 22 | ^X16 | ^0026 |
| 04 | 014F | 461 | .BYTE | ^X04 | 23 | ^X17 | ^0027 |
| 5C | 0150 | 462 | .BYTE | ^A/\ | 24 | ^X18 | ^0030 |
| 5C | 0151 | 463 | .BYTE | ^A/\ | 25 | ^X19 | ^0031 |
| 5C | 0152 | 464 | .BYTE | ^A/\ | 26 | ^X1A | ^0032 |
| 5C | 0153 | 465 | .BYTE | ^A/\ | 27 | ^X1B | ^0033 |
| 14 | 0154 | 466 | .BYTE | ^X14 | 28 | ^X1C | ^0034 |
| 15 | 0155 | 467 | .BYTE | ^X15 | 29 | ^X1D | ^0035 |
| 5C | 0156 | 468 | .BYTE | ^A/\ | 30 | ^X1E | ^0036 |
| 1A | 0157 | 469 | .BYTE | ^X1A | 31 | ^X1F | ^0037 |
| 30 | 0158 | 470 | .BYTE | ^A/O | 32 | ^X20 | ^0040 |
| 2F | 0159 | 471 | .BYTE | ^A/\ | 33 | ^X21 | ^0041 |
| 53 | 015A | 472 | .BYTE | ^A/S | 34 | ^X22 | ^0042 |
| 54 | 015B | 473 | .BYTE | ^A/T | 35 | ^X23 | ^0043 |
| 55 | 015C | 474 | .BYTE | ^A/U | 36 | ^X24 | ^0044 |
| 56 | 015D | 475 | .BYTE | ^A/V | 37 | ^X25 | ^0045 |
| 57 | 015E | 476 | .BYTE | ^A/W | 38 | ^X26 | ^0046 |
| 58 | 015F | 477 | .BYTE | ^A/X | 39 | ^X27 | ^0047 |
| 59 | 0160 | 478 | .BYTE | ^A/Y | 40 | ^X28 | ^0050 |
| 5C | 0161 | 479 | .BYTE | ^A/\ | 41 | ^X29 | ^0051 |
| 3B | 0162 | 480 | .BYTE | ^A/: | 42 | ^X2A | ^0052 |
| 2C | 0163 | 481 | .BYTE | ^A/; | 43 | ^X2B | ^0053 |
| 28 | 0164 | 482 | .BYTE | ^A/(| 44 | ^X2C | ^0054 |
| 22 | 0165 | 483 | .BYTE | ^A/' | 45 | ^X2D | ^0055 |
| 23 | 0166 | 484 | .BYTE | ^A/# | 46 | ^X2E | ^0056 |
| 25 | 0167 | 485 | .BYTE | ^A/% | 47 | ^X2F | ^0057 |
| 5A | 0168 | 486 | .BYTE | ^A/Z | 48 | ^X30 | ^0060 |
| 5C | 0169 | 487 | .BYTE | ^A/\ | 49 | ^X31 | ^0061 |
| 5C | 016A | 488 | .BYTE | ^A/\ | 50 | ^X32 | ^0062 |
| 5C | 016B | 489 | .BYTE | ^A/\ | 51 | ^X33 | ^0063 |
| 5C | 016C | 490 | .BYTE | ^A/\ | 52 | ^X34 | ^0064 |
| 0A | 016D | 491 | .BYTE | ^X0A | 53 | ^X35 | ^0065 |
| 17 | 016E | 492 | .BYTE | ^X17 | 54 | ^X36 | ^0066 |
| 1B | 016F | 493 | .BYTE | ^X1B | 55 | ^X37 | ^0067 |
| 5C | 0170 | 494 | .BYTE | ^A/\ | 56 | ^X38 | ^0070 |
| 5C | 0171 | 495 | .BYTE | ^A/\ | 57 | ^X39 | ^0071 |
| 5C | 0172 | 496 | .BYTE | ^A/\ | 58 | ^X3A | ^0072 |
| 5C | 0173 | 497 | .BYTE | ^A/\ | 59 | ^X3B | ^0073 |
| 5C | 0174 | 498 | .BYTE | ^A/\ | 60 | ^X3C | ^0074 |
| 05 | 0175 | 499 | .BYTE | ^X05 | 61 | ^X3D | ^0075 |
| 06 | 0176 | 500 | .BYTE | ^X06 | 62 | ^X3E | ^0076 |
| 07 | 0177 | 501 | .BYTE | ^X07 | 63 | ^X3F | ^0077 |
| 2D | 0178 | 502 | .BYTE | ^A/- | 64 | ^X40 | ^0100 |
| 4A | 0179 | 503 | .BYTE | ^A/J | 65 | ^X41 | ^0101 |
| 4B | 017A | 504 | .BYTE | ^A/K | 66 | ^X42 | ^0102 |
| 4C | 017B | 505 | .BYTE | ^A/L | 67 | ^X43 | ^0103 |
| 4D | 017C | 506 | .BYTE | ^A/M | 68 | ^X44 | ^0104 |
| 4E | 017D | 507 | .BYTE | ^A/N | 69 | ^X45 | ^0105 |
| 4F | 017E | 508 | .BYTE | ^A/O | 70 | ^X46 | ^0106 |
| 50 | 017F | 509 | .BYTE | ^A/P | 71 | ^X47 | ^0107 |
| 51 | 0180 | 510 | .BYTE | ^A/Q | 72 | ^X48 | ^0110 |
| 5C | 0181 | 511 | .BYTE | ^A/\ | 73 | ^X49 | ^0111 |
| 3A | 0182 | 512 | .BYTE | ^A/: | 74 | ^X4A | ^0112 |
| 24 | 0183 | 513 | .BYTE | ^A/S | 75 | ^X4B | ^0113 |

| | | | | | | | |
|----|------|-----|-------|--------|-----|------|-------|
| 2A | 0184 | 514 | .BYTE | ^A/*/ | 76 | ^X4C | ^0114 |
| 5B | 0185 | 515 | .BYTE | ^A/L/ | 77 | ^X4D | ^0115 |
| 3E | 0186 | 516 | .BYTE | ^A/>/ | 78 | ^X4E | ^0116 |
| 26 | 0187 | 517 | .BYTE | ^A/8/ | 79 | ^X4F | ^0117 |
| 52 | 0188 | 518 | .BYTE | ^A/R/ | 80 | ^X50 | ^0120 |
| 11 | 0189 | 519 | .BYTE | ^X11 | 81 | ^X51 | ^0121 |
| 12 | 018A | 520 | .BYTE | ^X12 | 82 | ^X52 | ^0122 |
| 13 | 018B | 521 | .BYTE | ^X13 | 83 | ^X53 | ^0123 |
| 5C | 018C | 522 | .BYTE | ^A/\ / | 84 | ^X54 | ^0124 |
| 5C | 018D | 523 | .BYTE | ^A/\ / | 85 | ^X55 | ^0125 |
| 08 | 018E | 524 | .BYTE | ^X08 | 86 | ^X56 | ^0126 |
| 5C | 018F | 525 | .BYTE | ^A/\ / | 87 | ^X57 | ^0127 |
| 18 | 0190 | 526 | .BYTE | ^X18 | 88 | ^X58 | ^0130 |
| 19 | 0191 | 527 | .BYTE | ^X19 | 89 | ^X59 | ^0131 |
| 5C | 0192 | 528 | .BYTE | ^A/\ / | 90 | ^X5A | ^0132 |
| 5C | 0193 | 529 | .BYTE | ^A/\ / | 91 | ^X5B | ^0133 |
| 1C | 0194 | 530 | .BYTE | ^X1C | 92 | ^X5C | ^0134 |
| 1D | 0195 | 531 | .BYTE | ^X1D | 93 | ^X5D | ^0135 |
| 1E | 0196 | 532 | .BYTE | ^X1E | 94 | ^X5E | ^0136 |
| 1F | 0197 | 533 | .BYTE | ^X1F | 95 | ^X5F | ^0137 |
| 7D | 0198 | 534 | .BYTE | ^A/3/ | 96 | ^X60 | ^0140 |
| 7E | 0199 | 535 | .BYTE | ^A/~/ | 97 | ^X61 | ^0141 |
| 73 | 019A | 536 | .BYTE | ^A/s/ | 98 | ^X62 | ^0142 |
| 74 | 019B | 537 | .BYTE | ^A/t/ | 99 | ^X63 | ^0143 |
| 75 | 019C | 538 | .BYTE | ^A/u/ | 100 | ^X64 | ^0144 |
| 76 | 019D | 539 | .BYTE | ^A/v/ | 101 | ^X65 | ^0145 |
| 77 | 019E | 540 | .BYTE | ^A/w/ | 102 | ^X66 | ^0146 |
| 78 | 019F | 541 | .BYTE | ^A/x/ | 103 | ^X67 | ^0147 |
| 79 | 01A0 | 542 | .BYTE | ^A/y/ | 104 | ^X68 | ^0150 |
| 5C | 01A1 | 543 | .BYTE | ^A/\ / | 105 | ^X69 | ^0151 |
| 5C | 01A2 | 544 | .BYTE | ^A/\ / | 106 | ^X6A | ^0152 |
| 5C | 01A3 | 545 | .BYTE | ^A/\ / | 107 | ^X6B | ^0153 |
| 5C | 01A4 | 546 | .BYTE | ^A/\ / | 108 | ^X6C | ^0154 |
| 5C | 01A5 | 547 | .BYTE | ^A/\ / | 109 | ^X6D | ^0155 |
| 5C | 01A6 | 548 | .BYTE | ^A/\ / | 110 | ^X6E | ^0156 |
| 5C | 01A7 | 549 | .BYTE | ^A/\ / | 111 | ^X6F | ^0157 |
| 7A | 01A8 | 550 | .BYTE | ^A/z/ | 112 | ^X70 | ^0160 |
| 5C | 01A9 | 551 | .BYTE | ^A/\ / | 113 | ^X71 | ^0161 |
| 5C | 01AA | 552 | .BYTE | ^A/\ / | 114 | ^X72 | ^0162 |
| 5C | 01AB | 553 | .BYTE | ^A/\ / | 115 | ^X73 | ^0163 |
| 5C | 01AC | 554 | .BYTE | ^A/\ / | 116 | ^X74 | ^0164 |
| 5C | 01AD | 555 | .BYTE | ^A/\ / | 117 | ^X75 | ^0165 |
| 5C | 01AE | 556 | .BYTE | ^A/\ / | 118 | ^X76 | ^0166 |
| 5C | 01AF | 557 | .BYTE | ^A/\ / | 119 | ^X77 | ^0167 |
| 5C | 01B0 | 558 | .BYTE | ^A/\ / | 120 | ^X78 | ^0170 |
| 5C | 01B1 | 559 | .BYTE | ^A/\ / | 121 | ^X79 | ^0171 |
| 5C | 01B2 | 560 | .BYTE | ^A/\ / | 122 | ^X7A | ^0172 |
| 5C | 01B3 | 561 | .BYTE | ^A/\ / | 123 | ^X7B | ^0173 |
| 5C | 01B4 | 562 | .BYTE | ^A/\ / | 124 | ^X7C | ^0174 |
| 5C | 01B5 | 563 | .BYTE | ^A/\ / | 125 | ^X7D | ^0175 |
| 5C | 01B6 | 564 | .BYTE | ^A/\ / | 126 | ^X7E | ^0176 |
| 5C | 01B7 | 565 | .BYTE | ^A/\ / | 127 | ^X7F | ^0177 |
| 2B | 01B8 | 566 | .BYTE | ^A/+/ | 128 | ^X80 | ^0200 |
| 41 | 01B9 | 567 | .BYTE | ^A/A/ | 129 | ^X81 | ^0201 |
| 42 | 01BA | 568 | .BYTE | ^A/B/ | 130 | ^X82 | ^0202 |
| 43 | 01BB | 569 | .BYTE | ^A/C/ | 131 | ^X83 | ^0203 |
| 44 | 01BC | 570 | .BYTE | ^A/D/ | 132 | ^X84 | ^0204 |

| | | | | | | | |
|----|------|-----|-------|-------|-----|------|-------|
| 45 | 01BD | 571 | .BYTE | ^A/E/ | 133 | ^X85 | ^0205 |
| 46 | 01BE | 572 | .BYTE | ^A/F/ | 134 | ^X86 | ^0206 |
| 47 | 01BF | 573 | .BYTE | ^A/G/ | 135 | ^X87 | ^0207 |
| 48 | 01C0 | 574 | .BYTE | ^A/H/ | 136 | ^X88 | ^0210 |
| 5C | 01C1 | 575 | .BYTE | ^A/I/ | 137 | ^X89 | ^0211 |
| 3F | 01C2 | 576 | .BYTE | ^A/J/ | 138 | ^X8A | ^0212 |
| 2E | 01C3 | 577 | .BYTE | ^A/K/ | 139 | ^X8B | ^0213 |
| 29 | 01C4 | 578 | .BYTE | ^A/L/ | 140 | ^X8C | ^0214 |
| 5D | 01C5 | 579 | .BYTE | ^A/M/ | 141 | ^X8D | ^0215 |
| 3C | 01C6 | 580 | .BYTE | ^A/N/ | 142 | ^X8E | ^0216 |
| 21 | 01C7 | 581 | .BYTE | ^A/O/ | 143 | ^X8F | ^0217 |
| 49 | 01C8 | 582 | .BYTE | ^A/P/ | 144 | ^X90 | ^0220 |
| 01 | 01C9 | 583 | .BYTE | ^X01 | 145 | ^X91 | ^0221 |
| 02 | 01CA | 584 | .BYTE | ^X02 | 146 | ^X92 | ^0222 |
| 03 | 01CB | 585 | .BYTE | ^X03 | 147 | ^X93 | ^0223 |
| 5C | 01CC | 586 | .BYTE | ^A/Q/ | 148 | ^X94 | ^0224 |
| 09 | 01CD | 587 | .BYTE | ^X09 | 149 | ^X95 | ^0225 |
| 5C | 01CE | 588 | .BYTE | ^A/R/ | 150 | ^X96 | ^0226 |
| FF | 01CF | 589 | .BYTE | ^XFF | 151 | ^X97 | ^0227 |
| 5C | 01D0 | 590 | .BYTE | ^A/S/ | 152 | ^X98 | ^0230 |
| 5C | 01D1 | 591 | .BYTE | ^A/T/ | 153 | ^X99 | ^0231 |
| 5C | 01D2 | 592 | .BYTE | ^A/U/ | 154 | ^X9A | ^0232 |
| 0B | 01D3 | 593 | .BYTE | ^X0B | 155 | ^X9B | ^0233 |
| 0C | 01D4 | 594 | .BYTE | ^X0C | 156 | ^X9C | ^0234 |
| 0D | 01D5 | 595 | .BYTE | ^X0D | 157 | ^X9D | ^0235 |
| 0E | 01D6 | 596 | .BYTE | ^X0E | 158 | ^X9E | ^0236 |
| 0F | 01D7 | 597 | .BYTE | ^X0F | 159 | ^X9F | ^0237 |
| 7B | 01D8 | 598 | .BYTE | ^A/V/ | 160 | ^XA0 | ^0240 |
| 61 | 01D9 | 599 | .BYTE | ^A/a/ | 161 | ^XA1 | ^0241 |
| 62 | 01DA | 600 | .BYTE | ^A/b/ | 162 | ^XA2 | ^0242 |
| 63 | 01DB | 601 | .BYTE | ^A/c/ | 163 | ^XA3 | ^0243 |
| 64 | 01DC | 602 | .BYTE | ^A/d/ | 164 | ^XA4 | ^0244 |
| 65 | 01DD | 603 | .BYTE | ^A/e/ | 165 | ^XA5 | ^0245 |
| 66 | 01DE | 604 | .BYTE | ^A/f/ | 166 | ^XA6 | ^0246 |
| 67 | 01DF | 605 | .BYTE | ^A/g/ | 167 | ^XA7 | ^0247 |
| 68 | 01E0 | 606 | .BYTE | ^A/h/ | 168 | ^XA8 | ^0250 |
| 5C | 01E1 | 607 | .BYTE | ^A/I/ | 169 | ^XA9 | ^0251 |
| 5C | 01E2 | 608 | .BYTE | ^A/J/ | 170 | ^XAA | ^0252 |
| 5C | 01E3 | 609 | .BYTE | ^A/K/ | 171 | ^XAB | ^0253 |
| 5C | 01E4 | 610 | .BYTE | ^A/L/ | 172 | ^XAC | ^0254 |
| 5C | 01E5 | 611 | .BYTE | ^A/M/ | 173 | ^XAD | ^0255 |
| 5C | 01E6 | 612 | .BYTE | ^A/N/ | 174 | ^XAE | ^0256 |
| 5C | 01E7 | 613 | .BYTE | ^A/O/ | 175 | ^XAF | ^0257 |
| 69 | 01E8 | 614 | .BYTE | ^A/i/ | 176 | ^XB0 | ^0260 |
| 5C | 01E9 | 615 | .BYTE | ^A/J/ | 177 | ^XB1 | ^0261 |
| 5C | 01EA | 616 | .BYTE | ^A/K/ | 178 | ^XB2 | ^0262 |
| 5C | 01EB | 617 | .BYTE | ^A/L/ | 179 | ^XB3 | ^0263 |
| 5C | 01EC | 618 | .BYTE | ^A/M/ | 180 | ^XB4 | ^0264 |
| 5C | 01ED | 619 | .BYTE | ^A/N/ | 181 | ^XB5 | ^0265 |
| 5C | 01EE | 620 | .BYTE | ^A/O/ | 182 | ^XB6 | ^0266 |
| 5C | 01EF | 621 | .BYTE | ^A/P/ | 183 | ^XB7 | ^0267 |
| 5C | 01F0 | 622 | .BYTE | ^A/Q/ | 184 | ^XB8 | ^0270 |
| 5C | 01F1 | 623 | .BYTE | ^A/R/ | 185 | ^XB9 | ^0271 |
| 5C | 01F2 | 624 | .BYTE | ^A/S/ | 186 | ^XBA | ^0272 |
| 5C | 01F3 | 625 | .BYTE | ^A/T/ | 187 | ^XBB | ^0273 |
| 5C | 01F4 | 626 | .BYTE | ^A/U/ | 188 | ^XBC | ^0274 |
| 5C | 01F5 | 627 | .BYTE | ^A/V/ | 189 | ^XBD | ^0275 |

| | | | | | | | |
|----|------|-----|-------|------|-----|------|-------|
| 5C | 01F6 | 628 | .BYTE | ^A/\ | 190 | ^XBE | ^0276 |
| 5C | 01F7 | 629 | .BYTE | ^A/\ | 191 | ^XBF | ^0277 |
| 7C | 01F8 | 630 | .BYTE | ^A/! | 192 | ^XC0 | ^0300 |
| 6A | 01F9 | 631 | .BYTE | ^A/j | 193 | ^XC1 | ^0301 |
| 6B | 01FA | 632 | .BYTE | ^A/k | 194 | ^XC2 | ^0302 |
| 6C | 01FB | 633 | .BYTE | ^A/l | 195 | ^XC3 | ^0303 |
| 6D | 01FC | 634 | .BYTE | ^A/m | 196 | ^XC4 | ^0304 |
| 6E | 01FD | 635 | .BYTE | ^A/n | 197 | ^XC5 | ^0305 |
| 6F | 01FE | 636 | .BYTE | ^A/o | 198 | ^XC6 | ^0306 |
| 70 | 01FF | 637 | .BYTE | ^A/p | 199 | ^XC7 | ^0307 |
| 71 | 0200 | 638 | .BYTE | ^A/q | 200 | ^XC8 | ^0310 |
| 5C | 0201 | 639 | .BYTE | ^A/\ | 201 | ^XC9 | ^0311 |
| 5C | 0202 | 640 | .BYTE | ^A/\ | 202 | ^XCA | ^0312 |
| 5C | 0203 | 641 | .BYTE | ^A/\ | 203 | ^XCB | ^0313 |
| 5C | 0204 | 642 | .BYTE | ^A/\ | 204 | ^XCC | ^0314 |
| 5C | 0205 | 643 | .BYTE | ^A/\ | 205 | ^XCD | ^0315 |
| 5C | 0206 | 644 | .BYTE | ^A/\ | 206 | ^XCE | ^0316 |
| 5C | 0207 | 645 | .BYTE | ^A/\ | 207 | ^XCF | ^0317 |
| 72 | 0208 | 646 | .BYTE | ^A/r | 208 | ^XD0 | ^0320 |
| 5C | 0209 | 647 | .BYTE | ^A/\ | 209 | ^XD1 | ^0321 |
| 5C | 020A | 648 | .BYTE | ^A/\ | 210 | ^XD2 | ^0322 |
| 5C | 020B | 649 | .BYTE | ^A/\ | 211 | ^XD3 | ^0323 |
| 5C | 020C | 650 | .BYTE | ^A/\ | 212 | ^XD4 | ^0324 |
| 5C | 020D | 651 | .BYTE | ^A/\ | 213 | ^XD5 | ^0325 |
| 5C | 020E | 652 | .BYTE | ^A/\ | 214 | ^XD6 | ^0326 |
| 5C | 020F | 653 | .BYTE | ^A/\ | 215 | ^XD7 | ^0327 |
| 5C | 0210 | 654 | .BYTE | ^A/\ | 216 | ^XD8 | ^0330 |
| 10 | 0211 | 655 | .BYTE | ^X10 | 217 | ^XD9 | ^0331 |
| 5C | 0212 | 656 | .BYTE | ^A/\ | 218 | ^XDA | ^0332 |
| 5C | 0213 | 657 | .BYTE | ^A/\ | 219 | ^XDB | ^0333 |
| 5C | 0214 | 658 | .BYTE | ^A/\ | 220 | ^XDC | ^0334 |
| 5C | 0215 | 659 | .BYTE | ^A/\ | 221 | ^XDD | ^0335 |
| 5C | 0216 | 660 | .BYTE | ^A/\ | 222 | ^XDE | ^0336 |
| 5C | 0217 | 661 | .BYTE | ^A/\ | 223 | ^XDF | ^0337 |
| 5C | 0218 | 662 | .BYTE | ^A/\ | 224 | ^XE0 | ^0340 |
| 5C | 0219 | 663 | .BYTE | ^A/\ | 225 | ^XE1 | ^0341 |
| 5C | 021A | 664 | .BYTE | ^A/\ | 226 | ^XE2 | ^0342 |
| 5C | 021B | 665 | .BYTE | ^A/\ | 227 | ^XE3 | ^0343 |
| 5C | 021C | 666 | .BYTE | ^A/\ | 228 | ^XE4 | ^0344 |
| 5C | 021D | 667 | .BYTE | ^A/\ | 229 | ^XE5 | ^0345 |
| 5C | 021E | 668 | .BYTE | ^A/\ | 230 | ^XE6 | ^0346 |
| 5C | 021F | 669 | .BYTE | ^A/\ | 231 | ^XE7 | ^0347 |
| 5C | 0220 | 670 | .BYTE | ^A/\ | 232 | ^XE8 | ^0350 |
| 5C | 0221 | 671 | .BYTE | ^A/\ | 233 | ^XE9 | ^0351 |
| 5C | 0222 | 672 | .BYTE | ^A/\ | 234 | ^XEA | ^0352 |
| 5C | 0223 | 673 | .BYTE | ^A/\ | 235 | ^XEB | ^0353 |
| 5C | 0224 | 674 | .BYTE | ^A/\ | 236 | ^XEC | ^0354 |
| 5C | 0225 | 675 | .BYTE | ^A/\ | 237 | ^XED | ^0355 |
| 5C | 0226 | 676 | .BYTE | ^A/\ | 238 | ^XEE | ^0356 |
| 5C | 0227 | 677 | .BYTE | ^A/\ | 239 | ^XEF | ^0357 |
| 5C | 0228 | 678 | .BYTE | ^A/\ | 240 | ^XF0 | ^0360 |
| 5C | 0229 | 679 | .BYTE | ^A/\ | 241 | ^XF1 | ^0361 |
| 5C | 022A | 680 | .BYTE | ^A/\ | 242 | ^XF2 | ^0362 |
| 5C | 022B | 681 | .BYTE | ^A/\ | 243 | ^XF3 | ^0363 |
| 5C | 022C | 682 | .BYTE | ^A/\ | 244 | ^XF4 | ^0364 |
| 5C | 022D | 683 | .BYTE | ^A/\ | 245 | ^XF5 | ^0365 |
| 5C | 022E | 684 | .BYTE | ^A/\ | 246 | ^XF6 | ^0366 |

CRDRIVER
V04-000

- CR11 CARD READER DRIVER

E 3

15-SEP-1984 23:42:03
5-SEP-1984 00:11:25

VAX/VMS Macro V04-00
[DRIVER.SRC]CRDRIVER.MAR;1

Page 13
(1)

| | | | | | | | | |
|----|------|-----|-------|------|---|-----|------|-------|
| 5C | 022F | 685 | .BYTE | ^A/\ | : | 247 | ^XF7 | ^0367 |
| 5C | 0230 | 686 | .BYTE | ^A/\ | : | 248 | ^XF8 | ^0370 |
| 5C | 0231 | 687 | .BYTE | ^A/\ | : | 249 | ^XF9 | ^0371 |
| 5C | 0232 | 688 | .BYTE | ^A/\ | : | 250 | ^XFA | ^0372 |
| 5C | 0233 | 689 | .BYTE | ^A/\ | : | 251 | ^XFB | ^0373 |
| 5C | 0234 | 690 | .BYTE | ^A/\ | : | 252 | ^XFC | ^0374 |
| 5C | 0235 | 691 | .BYTE | ^A/\ | : | 253 | ^XFD | ^0375 |
| 5C | 0236 | 692 | .BYTE | ^A/\ | : | 254 | ^XFE | ^0376 |
| 5C | 0237 | 693 | .BYTE | ^A/\ | : | 255 | ^XFF | ^0377 |


```
.SBTTL CR11 FUNCTION DECISION TABLE
0238 695
0238 696 : CR11 FUNCTION DECISION TABLE
0238 697 :
0238 698 :
0238 699 :
0238 700 CR_FUNCABLE:
0238 701 FUNCTAB >
0238 702 <READLBLK,-
0238 703 READPBLK,-
0238 704 READVBLK,-
0238 705 SENSEMODE,-
0238 706 SENSECHAR,-
0238 707 SETMODE,-
0238 708 SETCHAR,-
0238 709 >
0240 710 FUNCTAB >
0240 711 <READLBLK,-
0240 712 READPBLK,-
0240 713 READVBLK,-
0240 714 SENSEMODE,-
0240 715 SENSECHAR,-
0240 716 SETMODE,-
0240 717 SETCHAR,-
0240 718 >
0248 719 FUNCTAB CR READ,-
0248 720 <READLBLK,-
0248 721 READPBLK,-
0248 722 READVBLK,-
0248 723 >
0254 724 FUNCTAB +EXES$SETMODE,-
0254 725 <SETCHAR,-
0254 726 SETMODE,-
0254 727 >
0260 728 FUNCTAB +EXES$SENSEMODE,-
0260 729 <SENSECHAR,-
0260 730 SENSEMODE,-
0260 731 >
```

```
:FUNCTION DECISION TABLE
:LEGAL FUNCTION
:READ LOGICAL BLOCK
:READ PHYSICAL BLOCK
:READ VIRTUAL BLOCK
:SENSE READ MODE
:SENSE READER CHARACTERISTICS
:SET READER MODE
:SET READER CHARACTERISTICS
:
:BUFFERED I/O FUNCTIONS
:READ LOGICAL BLOCK
:READ PHYSICAL BLOCK
:READ VIRTUAL BLOCK
:SENSE READ MODE
:SENSE READER CHARACTERISTICS
:SET READER MODE
:SET READER CHARACTERISTICS
:
:READ FUNCTIONS
:READ LOGICAL BLOCK
:READ PHYSICAL BLOCK
:READ VIRTUAL BLOCK
:
:SET MODE/CHARACTERISTICS FUNCTIONS
:SET READER CHARACTERISTICS
:SET READER MODE
:
:SENSE MODE/CHARACTERISTICS FUNCTIONS
:SENSE READER CHARACTERISTICS
:SENSE READER MODE
:
```



```
026C 733 .SBTTL CANCEL I/O ON CHANNEL
026C 734 :+
026C 735 : CR_CANCELIO - CANCEL I/O ON CHANNEL
026C 736 :
026C 737 : THIS ROUTINE IS CALLED WHEN THE LAST CHANNEL ASSIGNED TO A DEVICE IS DEASSIGNED,
026C 738 : THE DEVICE IS DEALLOCATED, AND WHEN THE CANCEL I/O ON CHANNEL SYSTEM SERVICE IS
026C 739 : EXECUTED.
026C 740 :
026C 741 : INPUTS:
026C 742 :
026C 743 : R2 = NEGATIVE CHANNEL NUMBER.
026C 744 : R3 = ADDRESS OF CURRENT I/O REQUEST PACKET.
026C 745 : R4 = CURRENT PROCESS PCB ADDRESS.
026C 746 : R5 = DEVICE UCB ADDRESS.
026C 747 :
026C 748 : OUTPUTS:
026C 749 :
026C 750 : THE DEVICE INDEPENDENT CANCEL I/O ROUTINE IS CALLED AND A CHECK IS MADE
026C 751 : TO SEE IF THE UCB REFERENCE COUNT IS ZERO. IF THE REFERENCE COUNT IS ZERO,
026C 752 : THEN THE MESSAGE SENT TO JOB CONTROLLER BIT IS CLEARED.
026C 753 :-
026C 754 :
026C 755 CR_CANCELIO:
026C 756 TSTW UCBSW_REFC(R5) ;CANCEL I/O ON CHANNEL
026F 757 BNEQ 10$ ;REFERENCE COUNT ZERO?
0271 758 BICW #UCBSM_JOB,UCBSW_DEVSTS(R5) ;IF NEQ NO
0275 759 10$: JMP G*IOCS_CANCELIO ;CLEAR MESSAGE SENT BIT
;CANCEL I/O ON CHANNEL
```

5C A5 B5 026C 756 TSTW UCBSW_REFC(R5) ;CANCEL I/O ON CHANNEL
04 12 026F 757 BNEQ 10\$;REFERENCE COUNT ZERO?
68 A5 01 AA 0271 758 BICW #UCBSM_JOB,UCBSW_DEVSTS(R5) ;IF NEQ NO
0C000000 GF 17 0275 759 10\$: JMP G*IOCS_CANCELIO ;CLEAR MESSAGE SENT BIT
;CANCEL I/O ON CHANNEL


```

027B 761 .SBTTL READ FUNCTION PROCESSING
027B 762
027B 763 :+ CR_READ - READ FUNCTION PROCESSING
027B 764
027B 765 : THIS ROUTINE IS CALLED FROM THE FUNCTION DECISION TABLE DISPATCHER TO PROCESS
027B 766 : A READ LOGICAL, READ PHYSICAL, OR READ VIRTUAL FUNCTION TO A CARD READER.
027B 767
027B 768 : INPUTS:
027B 769
027B 770 : R0 = SCRATCH.
027B 771 : R1 = SCRATCH.
027B 772 : R2 = SCRATCH.
027B 773 : R3 = ADDRESS OF I/O REQUEST PACKET.
027B 774 : R4 = CURRENT PROCESS PCB ADDRESS.
027B 775 : R5 = ASSIGNED DEVICE UCB ADDRESS.
027B 776 : R6 = ADDRESS OF CCB.
027B 777 : R7 = I/O FUNCTION CODE.
027B 778 : R8 = FUNCTION DECISION TABLE DISPATCH ADDRESS.
027B 779 : R9 = SCRATCH.
027B 780 : R10 = SCRATCH.
027B 781 : R11 = SCRATCH.
027B 782 : AP = ADDRESS OF FIRST FUNCTION DEPENDENT PARAMETER.
027B 783
027B 784 : OUTPUTS:
027B 785
027B 786 : THE FUNCTION PARAMETERS ARE CHECKED AND A BUFFER IS ALLOCATED FOR THE
027B 787 : CARD READER DRIVER TO READ A CARD IMAGE INTO.
027B 788 :-
027B 789
027B 790 CR_READ:
027B 791 : READ FUNCTION PROCESSING
027B 792 : GET ADDRESS OF USER BUFFER
027E 793 : GET LENGTH OF USER BUFFER
0282 794 : IF EQL ZERO LENGTH TRANSFER
0284 795 : CHECK ACCESSIBILITY OF USER BUFFER
028A 796 : INSERT LENGTH OF USER BUFFER
028E 797 : SAVE BUFFER AND I/O PACKET ADDRESSES
0290 798 : SET LENGTH REQUIRED FOR ASCII READ
0294 799 : IF CLR, ASCII READ
0299 800 : SET LENGTH REQUIRED FOR BINARY READ
029C 801 : LENGTH OF READ LARGER THAN USER BUFFER?
02A0 802 : IF GEQU YES
02A2 803 : SET LENGTH OF USER BUFFER TO SIZE OF READ
02A6 804 : ACCOUNT FOR BUFFER OVERHEAD
02A9 805 : CHECK IF PROCESS HAS SUFFICIENT QUOTA
02AF 806 : IF LBC QUOTA CHECK FAILURE
02B2 807 : ALLOCATE BUFFER FOR CARD READ
02B8 808 : IF LBC ALLOCATION FAILURE
02BB 809 : RETRIEVE BUFFER AND I/O PACKET ADDRESSES
02BD 810 : INSERT ADDRESS OF READ BUFFER
02C1 811 : INSERT NUMBER OF QUOTA BYTES CHARGED
02C5 812 : SAVE BUFFER ADDRESS
02C7 813 : GET JIB ADDRESS
02CC 814 : CHARGE PROCESS FOR BUFFER
02D0 815 : RESTORE BUFFER ADDRESS
02D3 816 : INSERT ADDRESS OF DATA AREA
02D7 817 : SAVE ADDRESS OF USER BUFFER
02DA 817 : QUEUE DRIVER PACKET

```


| | | | | | | |
|----------|-----|------|------|-----|---|---|
| | | 02E0 | 818 | | | |
| | | 02E0 | 819 | : | | |
| | | 02E0 | 820 | : | ZERO LENGTH TRANSFER | |
| | | 02E0 | 821 | : | | |
| | | 02E0 | 822 | : | | |
| 50 | 01 | 3C | 02E0 | 823 | 30\$: | MOVZWL #SS\$ NORMAL, R0 ;SET NORMAL COMPLETION STATUS |
| 00000000 | 'GF | 17 | 02E3 | 824 | JMP G^EXE\$FINISHIOC ;FINISH I/O | |
| | | | 02E9 | 825 | | |
| | | | 02E9 | 826 | : | |
| | | | 02E9 | 827 | : | QUOTA OR BUFFER ALLOCATION FAILURE |
| | | | 02E9 | 828 | : | |
| | | | 02E9 | 829 | : | |
| | 0C | BA | 02E9 | 830 | 40\$: | POPR #^M<R2,R3> ;RETRIEVE I/O PACKET ADDRESS |
| 00000000 | 'GF | 17 | 02EB | 831 | JMP G^EXE\$ABORTIO ;ABORT I/O OPERATION | |


```
02F1 833 .SBTTL START I/O OPERATION ON CR11 CARD READER
02F1 834
02F1 835 :+ CR_STARTIO - START I/O OPERATION ON CR11 CARD READER
02F1 836
02F1 837 : THIS ROUTINE IS ENTERED WHEN THE ASSOCIATED UNIT IS IDLE AND A PACKET IS
02F1 838 : AVAILABLE FOR PROCESSING.
02F1 839
02F1 840 : INPUTS:
02F1 841
02F1 842 : R3 = ADDRESS OF I/O REQUEST PACKET.
02F1 843 : R5 = ADDRESS OF DEVICE UNIT UCB.
02F1 844
02F1 845 : OUTPUTS:
02F1 846
02F1 847 : CARD MOTION IS STARTED BY SETTING THE APPROPRIATE FUNCTION BITS IN THE
02F1 848 : CONTROL STATUS REGISTER. AS EACH COLUMN INTERRUPT OCCURS, THE DATA FROM
02F1 849 : THE DATA BUFFER REGISTER(S) IS STORED IN THE BUFFER ALLOCATED BY THE
02F1 850 : FDT ROUTINE. WHEN ALL 80 COLUMNS HAVE BEEN READ, A FORK PROCESS IS CREAT-
02F1 851 : ED, THE COLUMN DATA IS CONVERTED ACCORDING TO THE I/O FUNCTION CODE, AND
02F1 852 : REQUEST COMPLETE IS CALLED FOR POST PROCESSING.
02F1 853 :-
02F1 854
02F1 855 CR_STARTIO: : START I/O OPERATION
02F1 856 CMPZV #IRPSV_FCODE,#IRPSS_FCODE,- : SET MODE FUNCTION?
1A 06 00 ED 02F4 857 IRPSW_FUNC(R3),#IOS_SETMODE :
23 20 A3 13 02F7 858 BEQL 10$ : IF EQL YES
06 00 ED 02F9 859 CMPZV #IRPSV_FCODE,#IRPSS_FCODE,- : SET CHARACTERISTICS FUNCTION?
1A 20 A3 12 02FC 860 IRPSW_FUNC(R3),#IOS_SETCHAR :
0301 861 BNEQ 20$ : IF NEQ NO
0301 862
0301 863 : SET READER CHARACTERISTICS
0301 864
0301 865
0301 866
40 A5 38 A3 B0 0301 867 MOVW IRPSL_MEDIA(R3),UCBSB_DEVCLASS(R5) ; SET DEVICE CLASS AND TYPE
0306 868
0306 869 : SET READER MODE
0306 870
0306 871
0306 872
42 A5 3A A3 B0 0306 873 10$: MOVW IRPSL_MEDIA+2(R3),UCBSW_DEVBUFSIZ(R5) ; SET DEFAULT BUFFER SIZE
44 A5 3C A3 D0 030B 874 MOVL IRPSL_MEDIA+4(R3),UCBSL_DEVDEPEND(R5) ; SET DEVICE DEPENDENT FLAGS
016A 31 0310 875 BRW 140$
0313 876
0313 877 : SET UP PARAMETERS AND READ CARD
0313 878
0313 879
0313 880
78 A5 2C B3 D0 0313 881 20$: MOVL @IRPSL_SVAPTE(R3),UCBSL_SVAPTE(R5) ; SET ADDRESS OF BUFFER
0091 C5 01 90 0318 882 MOVB #1,UCBSB_CR_EOFcnt(R5) ; SET END OF FILE COUNT FOR ASCII
05 20 A3 06 E1 031D 883 BBC #IOSV_BINARY,IRPSW_FUNC(R3),30$ ; IF CLR, ASCII READ
0091 C5 08 90 0322 884 MOVB #8,UCBSB_CR_EOFcnt(R5) ; SET END OF FILE COUNT FOR BINARY
0092 C5 0091 C5 90 0327 885 30$: MOVB UCBSB_CR_EOFcnt(R5),UCBSB_CR_EOFcol(R5) ; SET REQUIRED NUMBER
0090 C5 01 8E 032E 886 MNEGB #1,UCBSB_CR_COLcnt(R5) ; SET INITIAL COLUMN COUNT
0093 C5 94 0333 887 CLRB UCBSB_CR_OFcnt(R5) ; SET INITIAL OFFLINE COUNT
54 24 A5 D0 0337 888 MOVL UCBSL_CRB(R5),R4 ; GET ADDRESS OF CRB
54 2C B4 D0 033B 889 MOVL @CRBSL_INTD+VECSL_IDB(R4),R4 ; GET DEVICE CSR ADDRESS
```



```
64 0100 8F B3 033F 890 40$: DSBINT ;DISABLE INTERRUPTS
5D 13 0345 891 BITW #CR_CSR_M_OFFLIN,CR_CSR(R4) ;READER OFFLINE?
034A 892 BEQL 70$ ;IF EQL NO
034C 893 WFIKPC 50$,#2 ;WAIT FOR TIMEOUT
0356 894 IOFORK ;CREATE FORK PROCESS
E1 11 035C 895 BRB 40$ ;
035E 896
035E 897
035E 898 : READER TIME OUT OR DEVICE ERROR
035E 899
035E 900
64 40 8F 9B 035E 901 50$: MOVZBW #CR_CSR_M_IE,CR_CSR(R4) ;CLEAR READER ERRORS
0362 902 SETIPL UCBSB_FTP(R5) ;LOWER TO DEVICE FORK LEVEL
22 64 A5 03 E0 0366 903 BBS #UCBSB_CANCEL,UCBSB_STS(R5),60$ ;IF SET, CANCEL I/O REQUESTED
FFCC 0093 C5 01 0F 9D 036B 904 ACBB #15,#1-UCBSB_CR_OFLEN(R5),40$ ;IF SET, NOT TIME FOR MESSAGE
0093 C5 94 0373 905 CLRB UCBSB_CR_OFLEN(R5) ;CLEAR OFFLINE COUNT
18 BB 0377 906 PUSHR #M<R3,R4> ;SAVE REGISTERS
54 05 9A 0379 907 MOVZBL #MSG$ DEVOFFLIN,R4 ;SET DEVICE MESSAGE NUMBER
53 00000000'GF 9E 037C 908 MOVAB G^SYS$GL_OPRMBX,R3 ;GET ADDRESS OF OPERATOR MAILBOX
00000000'GF 16 0383 909 JSB G^EXE$SNDEVMSG ;SEND MESSAGE TO OPERATOR
18 BA 0389 910 POPR #M<R3,R4> ;RESTORE REGISTERS
B2 11 038B 911 BRB 40$ ;
038D 912
038D 913 : CANCEL CURRENT READ REQUEST
038D 914
038D 915
038D 916
50 2C 3C 038D 917 60$: MOVZWL #SS$-ABORT,R0 ;SET ABORT STATUS
00F3 31 0390 918 BRW 150$ ;
0393 919
0393 920 : DATA OVERFLOW (MORE THAN 80 COL) DETECTED
0393 921
0393 922
64 40 8F 9B 0393 923 65$: MOVZBW #CR_CSR_M_IE,CR_CSR(R4) ;INHIBIT READS
0397 924 IOFORK ;
0082 C5 B6 039D 925 INCW UCBSB_ERRCNT(R5) ;INCREMENT HARDWARE ERROR COUNT
50 0054 8F 3C 03A1 926 MOVZWL #SS$-CTRLERR,R0 ;RETURN HARDWARE ERROR STATUS
00DD 31 03A6 927 BRW 150$ ;
03A9 928
03A9 929 : INITIATE READ
03A9 930
03A9 931
03A9 932
02 A4 B5 03A9 933 70$: TSTW CR_CRB1(R4) ;CLEAR COLUMN BUFFER
41 8F 9B 03AC 934 MOVZBW #CR_CSR_M_IE!CR_CSR_M_READ,- ;ENABLE INTERRUPTS AND START READ
64 03AF 935 CR_CSR(R4) ;
0380 936 80$: WFIKPC 50$,#3 ;WAITFOR INTERRUPT OR TIMEOUT
038A 937 BITW #CR_CSR_M_CRDONE!CR_CSR_M_ERROR,CR_CSR(R4) ;CARD DONE OR ERROR?
038F 938 BNEQ 120$ ;IF NEQ YES
51 02 A4 B0 03C1 939 MOVW CR_CRB1(R4),R1 ;READ BINARY COLUMN
52 04 A4 B0 03C5 940 MOVW CR_CRB2(R4),R2 ;READ PACKED COLUMN
0090 C5 96 03C9 941 INCB UCBSB_CR_COLCNT(R5) ;INCREMENT COLUMN COUNT
05 12 03CD 942 BNEQ 90$ ;IF NEQ NOT FIRST COLUMN
0094 C5 51 B0 03CF 943 MOVW R1,UCBSB_CR_FSTCOL(R5) ;SAVE FIRST COLUMN BINARY DATA
0090 C5 0092 C5 91 03D4 944 90$: CMPB UCBSB_CR_EOFCOL(R5),UCBSB_CR_COLCNT(R5) ;PAST END OF FILE DATA?
0B 15 03DB 945 BLEQ 100$ ;IF LEQ YES
51 0F0F 8F B1 03DD 946 CMPW #CR_EOF,R1 ;END OF FILE PUNCH?
```



```
0090 C5 0091 C5 04 12 03E2 947 BNEQ 100$ ;IF NEQ NO
50 8F 97 03E4 948 DECB UCBSB_CR_EOFcnt(R5) ;DECREMENT END OF FILE COUNT
07 78 B5 52 91 03E8 949 100$: CMPB #80,UCBSB_CR_COLcnt(R5) ;DATA OVERFLOW (MORE THAN 80 COL) ?
20 A3 06 E1 03F0 951 BLEQU 65$ ;IF LEQU, YES
78 B5 51 B0 03F4 952 MOVW R2,@UCBSL_SVAPTE(R5) ;STORE PACKED COLUMN
78 A5 D6 03FD 953 BBC #IOSV_BINARY,IRPSW_FUNC(R3),110$ ;IF CLR, ASCII READ
78 A5 D6 0400 954 MOVW R1,@UCBSL_SVAPTE(R5) ;STORE BINARY COLUMN
A5 11 0403 955 110$: INCL UCBSL_SVAPTE(R5) ;UPDATE BUFFER ADDRESS
0409 956 INCL UCBSL_SVAPTE(R5) ;UPDATE BUFFER ADDRESS
040B 957 DSBINT ;DISABLE INTERRUPTS
040B 958 BRB 80$ ;
040B 959 ;
040B 960 ; SPECIAL CONDITION
040B 961 ;
040B 962 ;
0096 C5 64 B0 040B 963 120$: MOVW CR_CSR(R4),UCBSW_CR_CSR(R5) ;SAVE READER STATUS
64 40 8F 9B 0410 964 MOVZBW #CR_CSR_M_IE,CR_CSR(R4) ;CLEAR READER ERRORS
50 0870 8F 3C 0414 965 IOFORK ;CREATE FORK PROCESS
0091 C5 95 041A 966 MOVZWL #SS$_ENDOFFILE,R0 ;ASSUME END OF FILE ENCOUNTERED
61 13 041F 967 TSTB UCBSB_CR_EOFcnt(R5) ;END OF FILE?
50 0838 8F 3C 0423 968 BEQL 150$ ;IF EQL YES
51 0096 C5 3C 0425 969 MOVZWL #SS$_DATAOVERUN,R0 ;ASSUME TIMING ERROR
53 51 0B E0 042A 970 UCBSW_CR_CSR(R5),R1 ;GET READER STATUS
65 51 0C E0 042F 971 BBS #CR_CSR_V_TIMERR,R1,150$ ;IF SET, TIMING ERROR - EXIT
0433 972 BBS #CR_CSR_V_MCHECK,R1,180$ ;IF SET, MOTION CHECK - RETRY
0437 973 ;
0437 974 ;*** NOTE: SINCE HOPPER CHECK SETS ERROR, A READ CHECK IS NOT DETECTABLE
0437 975 ;*** IF HOPPER CHECK IS ALSO SET, IE. NO READ CHECK RETRIES ON LAST CARD
0437 976 ;
04 51 0D E0 0437 977 BBS #CR_CSR_V_MCHECK,R1,125$ ;IF SET, HOPPER CHECK - OK
5D 51 0F E0 043B 978 BBS #CR_CSR_V_ERROR,R1,180$ ;IF SET, READ CHECK - RETRY
043F 979 125$: BITW #IOSM_BINARY!IOSM_PACKED,IRPSW_FUNC(R3) ;BINARY OR PACKED READ?
20 A3 00C0 8F B3 043F 980 BNEQ 140$ ;IF NEQ YES
0094 C5 0AAA 8F B1 0445 981 CMPW #CR_029,UCBSW_CR_FSTCOL(R5) ;CHANGE MODE TO 029 TRANSLATION?
3E 13 044E 982 BEQL 160$ ;IF EQL YES
0094 C5 0BA2 8F B1 0450 983 CMPW #CR_026,UCBSW_CR_FSTCOL(R5) ;CHANGE MODE TO 026 TRANSLATION?
3D 13 0457 984 BEQL 170$ ;IF EQL YES
50 FBDB CF 9E 0459 985 MOVAB CR_CVT029,R0 ;GET ADDRESS OF 029 TRANSLATION TABLE
04 00 ED 045E 986 CMPZV #CR$V_TMODE,#CR$S_TMODE,- ;029 TRANSLATION MODE?
01 44 A5 0461 987 UCBSL_DEVDEPEND(R5),#CR$K_1029 ;
05 13 0464 988 BEQL 130$ ;IF EQL YES
50 FCCE CF 9E 0466 989 MOVAB CR_CVT026,R0 ;GET ADDRESS OF 026 TRANSLATION TABLE
55 DD 046B 990 PUSHL R5 ;SAVE ADDRESS OF UCB
32 A3 60 00 51 2C B3 D0 046D 991 130$: MOVL @IRPSL_SVAPTE(R3),R1 ;GET ADDRESS OF I/O BUFFER
61 32 A3 2E 0471 992 MOVTC IRPSW_BCNT(R3),(R1),#0,(R0),IRPSW_BCNT(R3),(R1) ;TRANSLATE
55 BED0 047A 993 POPL R5 ;RETRIEVE ADDRESS OF UCB
50 10 10 50 01 3C 047D 994 140$: MOVZWL #SS$_NORMAL,R0 ;SET NORMAL COMPLETION
7E A5 F0 0480 995 INSV UCBSW_BCNT(R5),#16,#16,R0 ;INSERT TRANSFER BYTE COUNT
51 D4 0486 996 150$: CLRL R1 ;CLEAR SECOND I/O LONGWORD
0488 997 REQCOM ;COMPLETE REQUEST
048E 998 ;
048E 1000 ;
048E 1001 ; SET 029 TRANSLATION MODE
048E 1002 ;
```



```
00 01 F0 048E 1003
44 A5 04 048E 1004 160$: INSV #CR$K_T029,#CR$V_TMODE,-:SET 029 TRANSLATION MODE
06 11 0491 1005 #CR$S_TMODE,UCB$_DEVDEPEND(R5);
0494 1006 BRB 180$;
0496 1007
0496 1008 ::
0496 1009 :: SET 026 TRANSLATION MODE
0496 1010 ::
0496 1011
00 00 F0 0496 1012 170$: INSV #CR$K_T026,#CR$V_TMODE,-:SET 026 TRANSLATION MODE
44 A5 04 0499 1013 #CR$S_TMODE,UCB$_DEVDEPEND(R5);
FE74 31 049C 1014 180$: BRW 20$;
```



```
049F 1016 .SBTTL CR11 CARD READER INTDERRUPTS
049F 1017 :+
049F 1018 : CR$INT - CR11 CARD READER INTERRUPTS
049F 1019 :
049F 1020 : THIS ROUTINE IS ENTERED VIA A JSB INSTRUCTION WHEN AN INTERRUPT OCCURS ON A
049F 1021 : CR11 CARD READER CONTROLLER. THE STATE OF THE STACK ON ENTRY IS:
049F 1022 :
049F 1023 : 00(SP) = ADDRESS OF IDB ADDRESS.
049F 1024 : 04(SP) = 24(SP) = SAVED R0 - R5.
049F 1025 : 28(SP) = INTERRUPT PC.
049F 1026 : 32(SP) = INTERRUPT PSL.
049F 1027 :
049F 1028 : INTERRUPT DISPATCHING OCCURS AS FOLLOWS:
049F 1029 :
049F 1030 : IF THE INTERRUPT IS EXPECTED, THE DRIVER IS CALLED AT ITS
049F 1031 : INTERRUPT RETURN ADDRESS (UCB$FPC). IF THE INTERRUPT IS
049F 1032 : NOT EXPECTED AND THE DEVICE IS NOT ALLOCATED, A MESSAGE IS
049F 1033 : SENT TO THE JOB CONTROLLER TO INFORM IT THAT AN INPUT
049F 1034 : SYMBIONT PROCESS SHOULD BE CREATED TO READ THE CARDS.
049F 1035 : -
049F 1036 :
049F 1037 CR$INT:: : CARD READER INTERRUPT
11 64 53 9E D0 049F 1038 MOVL @ (SP)+,R3 : GET ADDRESS OF IDB
54 63 7D 04A2 1039 MOVQ IDB$ (CSR(R3),R4 : GET CONTROLLER CSR AND OWNER UCB ADDRESS
A5 01 E5 04A5 1040 BBCC #UCB$V INT,UCB$W_STS(R5) : 10$ : IF CLR, INTERRUPT NOT EXPECTED
53 10 A5 D0 04AA 1041 MOVL UCB$FPC(R5),R3 : RESTORE REMAINING DRIVER CONTEXT
OC B5 16 04AE 1042 JSB @UCB$FPC(R5) : CALL DRIVER
50 8E 7D 04B1 1043 MOVQ (SP)+,R0 : RESTORE REGISTERS
52 8E 7D 04B4 1044 MOVQ (SP)+,R2
54 8E 7D 04B7 1045 MOVQ (SP)+,R4
02 04BA 1046 REI
04BB 1047
04BB 1048 :
04BB 1049 : UNSOLICITED INTERRUPT
04BB 1050 :
04BB 1051 :
50 64 3C 04BB 1052 10$: MOVZWL CR_CSR(R4),R0 : GET READER STATUS
64 40 9B 04BE 1053 MOVZBW #CR_CSR_M_IE,CR_CSR(R4) : CLEAR STATUS, ENABLE INTERRUPTS
50 0400 8F B3 04C2 1054 BITW #CR_CSR_M_ONLINE,R0 : READER TRANSITION TO ONLINE?
OC 13 04C7 1055 BEQL 20$ : IF EQL NO
5C A5 B5 04C9 1056 TSTW UCB$W_REFC(R5) : DEVICE ASSIGNED OR ALLOCATED?
07 12 04CC 1057 BNEQ 20$ : IF NEQ YES
02 68 A5 00 E2 04CE 1058 BBSS #UCB$V_JOB,UCB$W_DEVSTS(R5),20$ : IF SET, MESSAGE ALREADY SENT
OA 10 04D3 1059 BSBB 30$ : SEND MESSAGE TO JOB CONTROLLER
50 8E 7D 04D5 1060 20$: MOVQ (SP)+,R0 : RESTORE REGISTERS
52 8E 7D 04D8 1061 MOVQ (SP)+,R2
54 8E 7D 04DB 1062 MOVQ (SP)+,R4
02 04DE 1063 REI
00000000 GF 16 04DF 1064 30$: JSB G*EXE$FORK : CREATE FORK PROCESS
54 02 9A 04E5 1065 MOVZBL #MSG$ CRUNSOLIC,R4 : SET MESSAGE TYPE
53 00000000 GF 9E 04E8 1066 MOVAB G*SYS$GL_JOBCTLMB,R3 : SET ADDRESS OF JOB CONTROLLER MAILBOX
00000000 GF 16 04EF 1067 JSB G*EXE$SNDEVMSG : SEND MESSAGE TO JOB CONTROLLER
04 50 E8 04F5 1068 BLBS R0,40$ : IF LBS SUCCESSFUL NOTIFICATION
68 A5 01 AA 04F8 1069 BICW #UCB$M_JOB,UCB$W_DEVSTS(R5) : CLEAR MESSAGE SENT BIT
05 04FC 1070 40$: RSB
```



```

04FD 1072 .SBTTL CARD READER INITIALIZATION
04FD 1073 :+
04FD 1074 : CR_INITIAL - CR11 CARD READER INITIALIZATION
04FD 1075 :
04FD 1076 : THIS ROUTINE IS CALLED AT SYSTEM STARTUP AND AFTER A POWER FAILURE. THE CSR
04FD 1077 : ADDRESS OF THE RESPECTIVE CR11 CONTROLLER IS READ TO INSURE ITS PRESENCE ON
04FD 1078 : THE UBA AND THEN CARD READER INTERRUPTS ARE ENABLED.
04FD 1079 :
04FD 1080 : INPUTS:
04FD 1081 :
04FD 1082 : R4 = CR11 CONTROLLER CSR ADDRESS.
04FD 1083 : R5 = IDB ADDRESS OF DEVICE UNIT.
04FD 1084 :
04FD 1085 : OUTPUTS:
04FD 1086 :
04FD 1087 : ALL REGISTERS ARE PRESERVED.
04FD 1088 :-
04FD 1089 :
04FD 1090 CR_INITIAL:
04FD 1091 MOVZBW #CR_CSR_M_IE,CR_CSR(R4) ;CR11 INITIALIZATION
04FD 1092 RSB ;ENABLE CR11 INTERRUPTS

```



```
0502 1094 .SBTTL CARD READER UNIT INITIALIZATION
0502 1095 :+
0502 1096 : CR_CR11_INIT - CARD READER UNIT INITIALIZATION
0502 1097 :
0502 1098 : THIS ROUTINE IS CALLED AT SYSTEM STARTUP AND AFTER A POWER FAILURE. THE
0502 1099 : ONLINE BIT IS SET IN THE DEVICE UCB.
0502 1100 :
0502 1101 : INPUTS:
0502 1102 :
0502 1103 : R5 = ADDRESS OF DEVICE UCB.
0502 1104 :
0502 1105 : OUTPUTS:
0502 1106 :
0502 1107 : THE ONLINE BIT IS SET IN THE DEVICE UCB AND THE ADDRESS OF THE UCB
0502 1108 : IS FILLED INTO THE OWNER FIELD OF THE IDB.
0502 1109 :-
0502 1110
0502 1111 CR_CR11_INIT:
0502 1112 BISW #UCB$M_ONLINE,UCB$W_STS(R5) ;CARD READER UNIT INITIALIZATION
0506 1113 MOVL UCB$L_CRB(R5),R0 ;SET UNIT ONLINE
050A 1114 MOVL CRB$L_INTD+VEC$L_IDB(R0),R0 ;GET ADDRESS OF CRB
050E 1115 MOVL R5,IDB$L_OWNER(R0) ;GET ADDRESS OF IDB
0512 1116 RSB ;SET ADDRESS OF DEVICE UCB
0513 1117 CR_END: ;ADDRESS OF LAST LOCATION IN DRIVER
0513 1118
0513 1119 .END
```

64 A5 10 A8
50 24 A5 D0
50 2C A0 D0
04 A0 55 D0
05

CRDRIVER
Symbol table

- CR11 CARD READER DRIVER

D 4

15-SEP-1984 23:42:03 VAX/VMS Macro V04-00
5-SEP-1984 00:11:25 [DRIVER.SRC]CRDRIVER.MAR;1

Page 25
(1)

| | | | | | | | |
|-----------------|------------|----|----|-------------------|------------|---|----|
| \$\$\$ | = 00000020 | R | 02 | EXESQIODRVPKT | ***** | X | 03 |
| \$\$OP | = 00000002 | | | EXESREADCHK | ***** | X | 03 |
| AT\$ UBA | = 00000001 | | | EXESSENSEMODE | ***** | X | 03 |
| CR\$DDT | = 00000000 | RG | 03 | EXESSETMODE | ***** | X | 03 |
| CR\$INT | = 0000049F | RG | 03 | EXESSNDEVMSG | ***** | X | 03 |
| CR\$K_TO26 | = 00000000 | | | FUNCTAB_LEN | = 00000034 | | |
| CR\$K_TO29 | = 00000001 | | | IDBSL_CSR | = 00000000 | | |
| CR\$S-TMODE | = 00000004 | | | IDBSL_OWNER | = 00000004 | | |
| CR\$V-TMODE | = 00000000 | | | IOSM_BINARY | = 00000040 | | |
| CRB\$C_INTD | = 00000024 | | | IOSM_PACKED | = 00000080 | | |
| CR_026 | = 000008A2 | | | IOSV_BINARY | = 00000006 | | |
| CR_029 | = 00000AAA | | | IOS_READBLK | = 00000021 | | |
| CR-CANCELIO | = 0000026C | R | 03 | IOS_READPBLK | = 0000000C | | |
| CR-CR11_INIT | = 00000502 | R | 03 | IOS_READVBLK | = 00000031 | | |
| CR-CRB1 | = 00000002 | | | IOS_SENSECHAR | = 0000001B | | |
| CR-CRB2 | = 00000004 | | | IOS_SENSEMODE | = 00000027 | | |
| CR_CSR | = 00000000 | | | IOS_SETCHAR | = 0000001A | | |
| CR_CSR_M_CRDONE | = 00004000 | | | IOS_SETMODE | = 00000023 | | |
| CR_CSR_M_ERROR | = 00008000 | | | IOS_VIRTUAL | = 0000003F | | |
| CR_CSR_M_IE | = 00000040 | | | IOCS-CANCELIO | ***** | X | 03 |
| CR_CSR_M_OFFLIN | = 00000100 | | | IOCSMNTVER | ***** | X | 03 |
| CR_CSR_M_ONLINE | = 00000400 | | | IOCSREQCOM | ***** | X | 03 |
| CR_CSR_M_READ | = 00000001 | | | IOCSRETURN | ***** | X | 03 |
| CR_CSR_V_ERROR | = 0000000F | | | IOCSWFIKPC | ***** | X | 03 |
| CR_CSR_V_MCHECK | = 0000000D | | | IRPSL_MEDIA | = 00000038 | | |
| CR_CSR_V_MCHECK | = 0000000C | | | IRPSL_SVAPTE | = 0000002C | | |
| CR_CSR_V_TIMERR | = 0000000B | | | IRPSS_FCODE | = 00000006 | | |
| CR_CVT026 | = 00000138 | R | 03 | IRPSV_FCODE | = 00000000 | | |
| CR_CVT029 | = 00000038 | R | 03 | IRPSW_BCNT | = 00000032 | | |
| CR_END | = 00000513 | R | 03 | IRPSW_BOFF | = 00000030 | | |
| CR_EOF | = 00000F0F | | | IRPSW_FUNC | = 00000020 | | |
| CR_FUNCTABLE | = 00000238 | R | 03 | JIBSL_BYTCNT | = 00000020 | | |
| CR_INITIAL | = 000004FD | R | 03 | MASKH | = 00000080 | | |
| CR_READ | = 0000027B | R | 03 | MASKL | = 08000000 | | |
| CR_STARTIO | = 000002F1 | R | 03 | MSG\$-CRUNSOLIC | = 00000002 | | |
| DC\$ CARD | = 00000041 | | | MSG\$-DEVOFFLIN | = 00000005 | | |
| DDB\$L_DDT | = 0000000C | | | P1 | = 00000000 | | |
| DEVSM_AVL | ***** | X | 02 | P2 | = 00000004 | | |
| DEVSM_IDV | ***** | X | 02 | P3 | = 00000008 | | |
| DEVSM_NNM | ***** | X | 02 | P4 | = 0000000C | | |
| DEVSM_REC | ***** | X | 02 | P5 | = 00000010 | | |
| DPTSC_LENGTH | = 00000038 | | | P6 | = 00000014 | | |
| DPTSC_VERSION | = 00000004 | | | PCBSL_JIB | = 00000080 | | |
| DPTSINITAB | = 00000038 | R | 02 | PRS_IPL | = 00000012 | | |
| DPTSREINITAB | = 00000062 | R | 02 | SIZ... | = 00000001 | | |
| DPTSTAB | = 00000000 | R | 02 | \$\$\$-ABORT | = 0000002C | | |
| DTS CR11 | = 00000001 | | | \$\$\$-CTRLERR | = 00000054 | | |
| DYN\$C-CRB | = 00000005 | | | \$\$\$-DATAOVERUN | = 00000838 | | |
| DYN\$C-DDB | = 00000006 | | | \$\$\$-ENDOFFILE | = 00000870 | | |
| DYN\$C-DPT | = 0000001E | | | \$\$\$-NORMAL | = 00000001 | | |
| DYN\$C-UCB | = 00000010 | | | SYSSGL_JOBCTLMB | ***** | X | 03 |
| EXESABORTIO | ***** | X | 03 | SYSSGL-OPRMBX | ***** | X | 03 |
| EXESALLOCBUF | ***** | X | 03 | UCBSB-CR_COLCNT | = 00000090 | | |
| EXESBUFFRQUOTA | ***** | X | 03 | UCBSB-CR_EOFcnt | = 00000091 | | |
| EXESFINISHIOC | ***** | X | 03 | UCBSB-CR_EOFCOL | = 00000092 | | |
| EXESFORK | ***** | X | 03 | UCBSB-CR_OFLCNT | = 00000093 | | |
| EXESIOFORK | ***** | X | 03 | UCBSB-DEVCLASS | = 00000040 | | |

CVD
V04

CRDRIVER
Symbol table

- CR11 CARD READER DRIVER

E 4

15-SEP-1984 23:42:03
5-SEP-1984 00:11:25

VAX/VMS Macro V04-00
[DRIVER.SRC]CRDRIVER.MAR;1

Page 26
(1)

```
UCBSB_DEVTYPE      = 00000041
UCBSB_DIPL         = 0000005E
UCBSB_FIPL         = 0000000B
UCBSK_CR_LENGTH    = 00000098
UCBSK_LENGTH       = 00000090
UCBSL_CRB          = 00000024
UCBSL_DEVCHAR      = 00000038
UCBSL_DEVCHAR2     = 0000003C
UCBSL_DEVDEPEND    = 00000044
UCBSL_FPC          = 0000000C
UCBSL_FR3          = 00000010
UCBSL_SVAPTE       = 00000078
UCBSM_JOB          = 00000001
UCBSM_ONLINE       = 00000010
UCBSV_CANCEL       = 00000003
UCBSV_INT          = 00000001
UCBSV_JOB          = 00000000
UCBSW_BCNT         = 0000007E
UCBSW_CR_CSR       = 00000096
UCBSW_CR_FSTCOL    = 00000094
UCBSW_DEVBUFSIZ    = 00000042
UCBSW_DEVSTS       = 00000068
UCBSW_ERRCNT       = 00000082
UCBSW_REFC         = 0000005C
UCBSW_STS          = 00000064
VECSL_IDB          = 00000008
VECSL_INITIAL      = 0000000C
VECSL_UNITINIT     = 00000018
```

! Psect synopsis !

| PSECT name | Allocation | PSECT No. | Attributes | | | | | | | | | | | | | | | | |
|--------------------|-------------------|-----------|------------|-----|-----|-----|-----|-------|-------|------|-------|-------|------|--|--|--|--|--|--|
| . ABS . | 00000000 (0.) | 00 (0.) | NOPIC | USR | CON | ABS | LCL | NOSHR | NOEXE | NORD | NOWRT | NOVEC | BYTE | | | | | | |
| \$AB\$\$ | 00000098 (152.) | 01 (1.) | NOPIC | USR | CON | ABS | LCL | NOSHR | EXE | RD | WRT | NOVEC | BYTE | | | | | | |
| \$\$\$105_PROLOGUE | 00000077 (119.) | 02 (2.) | NOPIC | USR | CON | REL | LCL | NOSHR | EXE | RD | WRT | NOVEC | BYTE | | | | | | |
| \$\$\$115_DRIVER | 00000513 (1299.) | 03 (3.) | NOPIC | USR | CON | REL | LCL | NOSHR | EXE | RD | WRT | NOVEC | LONG | | | | | | |

! Performance indicators !

| Phase | Page faults | CPU Time | Elapsed Time |
|------------------------|-------------|-------------|--------------|
| Initialization | 33 | 00:00:00.09 | 00:00:01.28 |
| Command processing | 122 | 00:00:00.41 | 00:00:02.79 |
| Pass 1 | 524 | 00:00:15.90 | 00:01:09.87 |
| Symbol table sort | 0 | 00:00:02.31 | 00:00:10.98 |
| Pass 2 | 209 | 00:00:03.35 | 00:00:12.59 |
| Symbol table output | 17 | 00:00:00.10 | 00:00:01.04 |
| Psect synopsis output | 2 | 00:00:00.02 | 00:00:00.02 |
| Cross-reference output | 0 | 00:00:00.00 | 00:00:00.00 |
| Assembler run totals | 909 | 00:00:22.18 | 00:01:38.63 |

The working set limit was 1800 pages.

125521 bytes (246 pages) of virtual memory were used to buffer the intermediate code.
There were 110 pages of symbol table space allocated to hold 2141 non-local and 29 local symbols.
1119 source lines were read in Pass 1, producing 18 object records in Pass 2.
36 pages of virtual memory were used to define 33 macros.

! Macro library statistics !

| Macro library name | Macros defined |
|--------------------------------------|----------------|
| ----- | ----- |
| -\$255\$DUA28:[SYS.OBJ]LIB.MLB;1 | 20 |
| -\$255\$DUA28:[SYS.LIB]STARLET.MLB;2 | 11 |
| TOTALS (all libraries) | 31 |

2388 GETS were required to define 31 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:CRDRIVER/OBJ=OBJ\$:CRDRIVER MSRC\$:CRDRIVER/UPDATE=(ENH\$:CRDRIVER)+EXECMLS/LIB

0108 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

